ELSEVIER Journal of Computational and Applied Mathematics 82 (1997) 407–422

JOURNAL OF
COMPUTATIONAL AND
APPLIED MATHEMATICS

Parallel linear system solvers for Runge–Kutta–Nyström methods

P.J. van der Houwen[a,*], E. Messina[b]

[a] CWI, P.O. Box 94079, 1090 GB Amsterdam, Netherlands

[b] Dipartimento di Matematica e Applicazioni "R. Caccioppoli", University of Napels "Federico II", Via Cintia, I-80126 Napels, Italy

Received 2 September 1996; received in revised form 15 December 1996

Abstract

Solving the nonlinear systems arising in implicit Runge–Kutta–Nyström type methods by (modified) Newton iteration leads to linear systems whose matrix of coefficients is of the form $I - A \otimes h^2 J$ where A is the Runge–Kutta–Nyström matrix and J an approximation to the Jacobian of the right-hand-side function of the system of differential equations. For larger systems of differential equations, the solution of these linear systems by a direct linear solver is very costly, mainly because of the LU-decomposition. We try to reduce these costs by solving the linear Newton systems by an inner iteration process. Each inner iteration again requires the solution of a linear system. However, the matrix of coefficients in these new linear systems are of the form $I - B \otimes h^2 J$ where B is a nondefective matrix with positive eigenvalues, so that by a similarity transformation, we can decouple the system into subsystems the dimension of which equals the dimension of the system of differential equations. Since the subsystems can be solved in parallel, the resulting integration method is highly efficient on parallel computer systems. The performance of the *parallel iterative linear system* method for Runge–Kutta–Nyström equations (PILSRKN method) is illustrated by means of a few examples from the literature.

Keywords: Numerical analysis; Convergence of iteration methods; Runge–Kutta methods; Parallelism

1. Introduction

Suppose that we integrate the initial-value problem (IVP) for the system of special second-order equations

$$\frac{d^2y}{dt^2} = f(y), \quad y, f \in \mathbb{R}^d \tag{1.1}$$

PII S0377-0427(97)00050-2

by the Runge–Kutta–Nyström (RKN) method

$$y_n = y_{n-1} + hy'_{n-1} + h^2(\boldsymbol{b}^{\mathrm{T}} \otimes I)F(Y_n), \quad y'_n = y'_{n-1} + h(\boldsymbol{d}^{\mathrm{T}} \otimes I)F(Y_n) \tag{1.2}$$

where the stage vector Y_n is the solution of the equation

$$R(Y_n) = \boldsymbol{0}, \qquad R(Y) := Y - h^2(A \otimes I)F(Y) - \boldsymbol{e} \otimes y_{n-1} - h\boldsymbol{c} \otimes y'_{n-1}. \tag{1.3}$$

This equation will be referred to as the corrector equation. In the RKN method $\{(1.2),(1.3)\}$, A denotes a nonsingular $s \times s$ matrix, $\boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e}$ are s-dimensional vectors, $\boldsymbol{e}$ being the vector with unit entries, h is the stepsize $t_n - t_{n-1}, \otimes$ denotes the Kronecker product, and I is the $d \times d$ identity matrix (in the following, we shall use the notation I for any identity matrix, however, its order will always be clear from the context). The s components Y_{ni} of the sd-dimensional stage vector Y_n represent s numerical approximations to the s exact solution vectors $y(t_{n-1} + c_i h)$, where $\boldsymbol{c} = (c_i)$ denotes the abscissa vector. It is assumed that the components of $\boldsymbol{c}$ are distinct. Furthermore, for any vector $Y = (Y_i), F(Y)$ contains the derivative values $(f(Y_i))$. The arrays $\{A, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}\}$ define the RKN method. In this paper, we shall confine our considerations to RKN methods that originate from RK methods, that is, if the RK method is defined by the triple $\{A_{\mathrm{RK}}, \boldsymbol{b}_{\mathrm{RK}}, \boldsymbol{c}\}$ then the corresponding RKN method is defined by $\{(A_{\mathrm{RK}})^2, A_{\mathrm{RK}}^{\mathrm{T}}\boldsymbol{b}_{\mathrm{RK}}, \boldsymbol{c}, \boldsymbol{b}_{\mathrm{RK}}\}$ (see, [3]).

In the following, the Jacobian $J := \partial f(y)/\partial y$ of $f(y)$ is assumed to have a negative spectrum (that is, the IVP for (1.1) is assumed to be stable). Since we want to apply the RKN method to problems where J may have large, negative eigenvalues (such problems will be called *stiff* IVPs), we shall use the Shampine type-step point formulas, i.e. we rewrite (1.2) as (cf. [12], see also [5, p. 129])

$$\begin{aligned} y_n &= y_{n-1} + hy'_{n-1} + (\boldsymbol{b}^{\mathrm{T}}A^{-1} \otimes I)(Y_n - \boldsymbol{e} \otimes y_{n-1} - h\boldsymbol{c} \otimes y'_{n-1}), \\ y'_n &= y'_{n-1} + h^{-1}(\boldsymbol{d}^{\mathrm{T}}A^{-1} \otimes I)(Y_n - \boldsymbol{e} \otimes y_{n-1} - h\boldsymbol{c} \otimes y'_{n-1}). \end{aligned} \tag{1.4}$$

In actual implementation, these (algebraically equivalent) formulas are much more stable than (1.2). The conventional way of solving the corrector Eq. (1.3) is the modified Newton iteration scheme. In the case of Runge–Kutta methods, we developed in [8] a parallel linear solver for the solution of the linear systems that arise in each modified Newton iteration. In the present paper, we investigate how this linear solver should be adapted in the case of RKN methods.

2. A parallel linear solver

Application of modified Newton iteration to the corrector Eq. (1.3) yields

$$(I - A \otimes h^2 J)(Y_n^{(j)} - Y_n^{(j-1)}) = -R(Y_n^{(j-1)}), \quad j = 1, 2, \ldots, m, \tag{2.1}$$

where J is evaluated at t_n and $Y_n^{(0)}$ is the initial iterate to be provided by some predictor formula. Each Newton iteration requires the solution of an sd-dimensional linear system for the Newton correction $Y_n^{(j)} - Y_n^{(j-1)}$. If the linear systems in (2.1) are solved by a direct linear solver, then the bulk of the computational effort often goes in the LU-decomposition of the matrix $I - A \otimes h^2 J$. In the case of (2.1) this would mean the LU-decomposition of an $sd \times sd$ matrix requiring $O(s^3 d^3)$ arithmetic operations.

In order to achieve a reduction of the computational complexity of the process (2.1), we introduce an iterative method for solving the linear systems in (2.1). Following [8], this *inner* iteration process reads:

$$\begin{aligned}&(I - B \otimes h^2 J)(Y_n^{(j,v)} - Y_n^{(j,v-1)}) = -(I - A \otimes h^2 J) Y_n^{(j,v-1)} + C_n^{(j-1)}, \\ &C_n^{(j-1)} := (I - A \otimes h^2 J) Y_n^{(j-1)} - \boldsymbol{R}(Y_n^{(j-1)}),\end{aligned} \qquad v = 1,2,\ldots,r, \qquad (2.2)$$

where $Y_n^{(j,0)} = Y_n^{(j-1,r)}$ and where $Y_n^{(m,r)}$ is accepted as the solution Y_n of the corrector Eq. (1.3). Furthermore, B is a nondefective, real matrix with positive eigenvalues, and hence diagonalizable. The iterative method $\{(2.1),(2.2)\}$ may be considered as an outer–inner iteration process where the modified Newton iteration represents the outer iteration. Note that $C^{(j-1)}$ does not depend on v, so that the application of the inner iteration process requires only one evaluation of the function $\boldsymbol{R}$.

Since B is assumed to be diagonalizable, we may write $B = S\tilde{B}S^{-1}$ with S a real matrix and $\tilde{B}$ a diagonal matrix whose diagonal entries are the eigenvalues of B. By performing a similarity transformation $Y_n^{(j,v)} = (S \otimes I) X^{(j,v)}$ (cf. [1]), the process (2.2) transforms to

$$\begin{aligned}(I - \tilde{B} \otimes h^2 J)(X^{(j,v)} - X^{(j,v-1)}) = -(I - S^{-1}AS \otimes h^2 J) X^{(j,v-1)} + (S^{-1} \otimes I) C_n^{(j-1)}, \\ v = 1,\ldots,r,\end{aligned} \qquad (2.3)$$

where $X^{(j,0)} = (S^{-1} \otimes I) Y_n^{(j-1)}$. If for a given j, the transformed inner iterates $X^{(j,v)}$ converge to a vector $X^{(j,\infty)}$, then the modified Newton iterate defined by (2.1) can be obtained from $Y_n^{(j)} = (S \otimes I) X^{(j,\infty)}$. The iterations in (2.3) are diagonal-implicit, so that the LU-decomposition of the matrix $I - \tilde{B} \otimes h^2 J$ splits into s LU-decompositions of dimension d which can all be computed in parallel. Thus, the LU costs associated with (2.3) are a factor s^2 less than the LU costs associated with (2.1), and effectively (on an s-processor system) even a factor s^3.

As to the total computational effort of the modified Newton process (2.1) and the outer–inner iteration process $\{(2.1),(2.3)\}$, we remark that on top of the updates of the Jacobian matrix J and the LU-decomposition of the linear system matrices, the modified Newton process requires m forward–backward substitutions of dimension sd, whereas the outer–inner iteration process requires mrs forward–backward substitutions of dimension d. However, in the case of (2.3), the forward–backward substitutions can be distributed over s processors.

We shall call (2.3) a Parallel Iterative Linear System solver for RKN methods (PILSRKN method). Given the matrix A, it is completely defined by the matrices $\tilde{B}$ and S.

3. Convergence of the iterative linear solver

The speed of convergence of the method $\{(2.1),(2.3)\}$ depends on the modified Newton iteration process (2.1) and the inner iteration process (2.3). In general, modified Newton converges relatively fast, and usually only a few iterations suffice to solve the corrector Eq. (1.3). The convergence of the inner iteration process (2.3) is highly dependent on the matrices $\tilde{B}$ and S. This will be the subject of the following subsections.

3.1. Convergence region of the inner iteration process

In order to analyse the region of convergence for the inner iteration process, we consider the error recursion

$$Y_n^{(j,\nu)} - Y_n^{(j)} = M(Y_n^{(j,\nu-1)} - Y_n^{(j)}), \qquad M := (I - B \otimes h^2 J)^{-1}((A - B) \otimes h^2 J). \tag{3.1}$$

We have convergence if the powers M^ν of the amplification matrix M tend to zero as $\nu \to \infty$, that is, if the spectral radius $\rho(M)$ of M is less than 1. Consider the vectors $\boldsymbol{a} \otimes \boldsymbol{w}$, where $\boldsymbol{w}$ is an eigenvector of J and $\boldsymbol{a}$ is an eigenvector of the matrix

$$Z(x) := x(I - xB^{-1})(A - B), \quad x := h^2\lambda \tag{3.2}$$

with λ in the eigenspectrum $\sigma(J)$ of J (we recall that J is assumed to have a negative spectrum of frequencies λ, otherwise, the IVP for (1.1) would be unstable). Evidently, these vectors are eigenvectors of M with eigenvalues given by the eigenvalues of $Z(h^2\lambda)$. Suppose that the Jacobian matrix J and the matrix $Z(h^2\lambda)$ with $\lambda \in \sigma(J)$ both have a complete eigensystem. Then, M has sd eigenvectors of the form $\boldsymbol{a} \otimes \boldsymbol{w}$, and hence, *all* its eigenvalues are given by those of the matrix $Z(h^2\lambda)$ with $\lambda \in \sigma(J)$. This justifies to define $\Gamma := \{x\colon \rho(Z(x)) < 1, x \leqslant 0\}$ as the *interval of convergence* of the inner iteration process. Thus, we have convergence if the eigenvalues of h^2J lie in Γ. If Γ contains the whole nonpositive real axis, then the inner iteration process will be called A_0*-convergent.*

We shall call $Z(x)$ the *amplification matrix* at the point x and $\rho(Z(x))$ the (*asymptotic*) *amplification factor* at x. The maximal amplification factor, i.e. the supremum of $\rho(Z(x))$ on the nonpositive axis, will be denoted by $\rho_{\max}$. Furthermore, we define the (*averaged*) *amplification factor*

$$\rho^{(\nu)} := \max\{\rho^{(\nu)}(x)\colon x \leqslant 0\}, \qquad \rho^{(\nu)}(x) := \sqrt[\nu]{\|Z^\nu(x)\|}. \tag{3.3}$$

Note that $\rho^{(\nu)}(x)$ approximates the asymptotic amplification factor $\rho(Z(x))$ as $\nu \to \infty$.

Since, it seems not feasible to minimize $\|Z(x)\|$ over all possible (real, nondefective) matrices B with positive eigenvalues, we decided to follow an alternative approach. Obviously, we may write $B = Q\tilde{T}Q^{-1}$ where Q is a nonsingular, real matrix and $\tilde{T}$ is a lower triangular matrix with positive diagonal entries. By performing the similarity transformation $Y_n^{(j,\nu)} = (Q \otimes I)\tilde{Y}_n^{(j,\nu)}$, the process (2.2) can be transformed to

$$\begin{aligned}(I - \tilde{T} \otimes h^2J)(\tilde{Y}_n^{(j,\nu)} - \tilde{Y}_n^{(j,\nu-1)}) = -(I - \tilde{A} \otimes h^2J)\tilde{Y}_n^{(j,\nu-1)} + (Q^{-1} \otimes I)C_n^{(j-1)},\\ \nu = 1, 2, \ldots, r,\end{aligned} \tag{3.4}$$

where $\tilde{A} := Q^{-1}AQ$ and $\tilde{Y}_n^{(j,0)} = (Q^{-1} \otimes I)Y_n^{(j-1)}$. The iteration process (3.4) will not be used in an actual implementation, but only serves to construct a suitable matrix B. We shall specify special families of matrix pairs $(\tilde{T}, Q)$ and perform a minimization process for the asymptotic amplification factor $\rho_{\max}$ within these families. The derivation of suitable families of matrices B can be based on the observation that strong damping of the *stiff* error components usually ensures a fast overall convergence (for a detailed discussion of this aspect, we refer to [6]). Here, *stiff* error components are understood to be components corresponding to eigenvectors of J with eigenvalues λ of large magnitude. This leads us to require the matrix $\tilde{T}$ to be such that $\rho(Z(x))$ is small at infinity. The next result is similar to a result derived in [7] and covers this situation:

Theorem 3.1. *Let Q be an arbitrary, nonsingular matrix and let $\tilde{A} := Q^{-1}AQ$ have the Crout decomposition $\tilde{A} = LU$, where L and U are, respectively, lower triangular and unit upper triangular. Then, the asymptotic amplification factor vanishes at infinity if $\tilde{T} = L$.*

Proof. It follows from the representation for $\tilde{A}$ that

$$Q^{-1}Z(\infty)Q = -\tilde{T}^{-1}(\tilde{A} - \tilde{T}) = -\tilde{T}^{-1}L(U - L^{-1}\tilde{T}).$$

By setting $\tilde{T} = L$, we achieve that $Q^{-1}Z(\infty)Q = I - U$ which is strictly upper triangular so that $\rho(Q^{-1}Z(\infty)Q) = \rho(Z(\infty)) = 0$. □

Theorem 3.1 defines a family of PILSRKN methods satisfying $\rho(Z(\infty)) = \rho(I - B^{-1}A) = 0$. In the construction of families of suitable transformation matrices Q, our guide line will be to increase the lower-triangular dominance of the matrix $\tilde{A} := Q^{-1}AQ$. In the following subsections we discuss three options. The matrix B and the corresponding vector of amplification factors $\rho = (\rho^{(\nu)})$ resulting from these options (with respect to the Euclidean norm) will be explicitly computed for the RKN corrector generated by the four-stage Radau IIA method. Details for RKN correctors generated by other RK methods will be given in [10].

3.2. Diagonal transformation matrices

The most simple family of transformation matrices is formed by the nonsingular, diagonal matrices $Q = D$ leading to $\tilde{A} := D^{-1}AD$ and $\tilde{T} := D^{-1}BD$. At first sight, it seems that the effectiveness of the matrix B is increased by choosing D such that the upper triangular part of $\tilde{A}$ has entries of small magnitude. However, that need not to be the case. For example, if we choose $\tilde{T}$ according to Theorem 3.1, then $B = DLD^{-1}$, where L satisfies $LU = D^{-1}AD$ with U unit upper triangular. Hence, we have the relation $DLD^{-1}DUD^{-1} = A$. Since DUD^{-1} is again unit upper triangular, DLD^{-1} turns out to be the lower triangular Crout factor of A. Thus, B does not depend on D, so that we may equally well set $D = I$. Similarly, if we identify $\tilde{T}$ with the lower triangular part of $\tilde{A}$, we obtain a matrix B that does not depend on D.

Calculations for a number of Gauss–Legendre and Radau IIA correctors with $Q = I$ and $\tilde{T}$ defined according to Theorem 3.1 will be reported in [10]. These calculations show that $\tilde{T}$ does have positive diagonal entries and generates A_0-convergent PILSRKN methods. For the four-stage Radau IIA corrector we found

$$B = \tilde{T} = \begin{pmatrix} 0.00672834 & 0 & 0 & 0 \\ 0.06814566 & 0.08355843 & 0 & 0 \\ 0.15530325 & 0.28718085 & 0.11595801 & 0 \\ 0.20093191 & 0.41620407 & 0.24088357 & 0.02173913 \end{pmatrix}. \qquad (3.5)$$

We remark that there is no need to implement the linear solver with a high precision matrix B, because the amplification factors will not change much. In the case (3.5) the amplification vector is given by

$$\rho = (1.62, 1.07, 0.75, 0.71, \ldots, 0.63). \qquad (3.6)$$

Thus, convergence starts in the third iteration. However, we should bear in mind that the amplifications factors $\rho^{(\nu)}$ are "worst case" values, so that in many problems, convergence may start already in the second or first iteration.

3.3. *Transformation to block-triangular form*

In [8] where the RK case has been investigated, the matrix Q was chosen such that $\tilde{A} = Q^{-1}AQ$ becomes a (real) σ-by-σ lower block-triangular matrix $\tilde{A} = (\tilde{A}_{kl})$, of which the diagonal blocks $\tilde{A}_{kk}$ are either 1×1 or 2×2 matrices. In some sense, this is the "best" we can achieve in the lower-triangularization of $\tilde{A}$. At the same time, this class of transformation matrices allows us to minimize the asymptotic amplification factor $\rho_{\max}$ by analytical means and to prove A_0-convergence. Following [8], we set $\tilde{A}_{kk} = \xi_k$ if ξ_k a real eigenvalue of A, and we set

$$\tilde{A}_{kk} = \begin{pmatrix} a_k & b_k \\ c_k & 2\xi_k - a_k \end{pmatrix}, \quad b_k = -c_k^{-1}(a_k^2 - 2\xi_k a_k + \alpha_k^2), \quad c_k \neq 0, \quad \alpha_k := \sqrt{\xi_k^2 + \eta_k^2}, \tag{3.7}$$

if $\xi_k \pm i\eta_k$ is a complex eigenvalue pair of A. Here, a_k and c_k are free parameters. Let K denote the set of integers with the property that $\eta_k \neq 0$ whenever $k \in K$. Then, a natural choice for $\tilde{T}$ now is

$$\tilde{T} := \begin{pmatrix} \tilde{T}_{11} & \text{O} & \text{O} & \text{O} & \cdots \\ \tilde{A}_{21} & \tilde{T}_{22} & \text{O} & \text{O} & \cdots \\ \tilde{A}_{31} & \tilde{A}_{32} & \tilde{T}_{33} & \text{O} & \cdots \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{pmatrix}, \quad \tilde{T}_{kk} := \begin{pmatrix} u_k & 0 \\ v_k & w_k \end{pmatrix} \quad \text{if } k \in K, \ \tilde{T}_{kk} = \xi_k \quad \text{if } k \notin K, \tag{3.8}$$

where u_k, v_k and w_k are free parameters with u_k and w_k assumed to be positive.

3.3.1. *A_0 convergent methods*

In this subsection, we try to construct matrices $\tilde{T}$ such that the generated PILSRKN method is A_0-convergent. Note that the A_0-convergence does not depend on Q. Recalling that we want strong damping of the stiff error components, we may resort to Theorem 3.1 and choose $\tilde{T}$ such that it becomes the lower triangular Crout factor of $\tilde{A}$. However, we can proceed slightly more generally by deriving the complete set of matrices $\tilde{T}$ leading to a vanishing asymptotic amplification factor $\rho(Z(\infty))$. Within this set we shall look for the matrix $\tilde{T}$ yielding a minimal asymptotic amplification factor $\rho_{\max}$.

Theorem 3.2. *Let A have its eigenvalues in the positive halfplane, let Q satisfy $\tilde{A} = Q^{-1}AQ$ where the diagonal blocks of $\tilde{A}$ are defined by* (3.7) *and let $\tilde{T}$ be defined by* (3.8) *with*

$$u_k = \gamma_k \alpha_k, \quad v_k = -c_k \frac{a_k \alpha_k + \gamma_k^2 \alpha_k (2\xi_k - a_k) - 2\gamma_k \alpha_k^2}{\gamma_k (a_k^2 - 2\xi_k a_k + \alpha_k^2)}, \quad w_k = \frac{\alpha_k}{\gamma_k}, \quad k \in K, \tag{3.9}$$

where $\gamma_k > 0$. Then, for all a_k and c_k, the following assertions hold for the PILSRKN method:

(i) $\rho(Z(\infty)) = 0$.

(ii) *The eigenvalues of B are positive.*

(iii) *It is A_0-convergent with* $\rho_{\max} = \max\{|1 - 2\gamma_k(\gamma_k + 1)^{-2}(\alpha_k + \xi_k)\alpha_k^{-1}|\colon k \in K\}$.
(iv) *If* $\tilde{A}$ *is block-diagonal, then* $\rho^{(v)}(x) = O(x^{(1-v)/v})$ *as* $x \to \infty$.

Proof. If $\tilde{T}$ is of the form (3.8), then the value of $\rho(Z(x))$ equals the maximum of the spectral radius $\rho(\tilde{Z}_{kk}(x))$ of the diagonal blocks $\tilde{Z}_{kk} := x(1 - x\tilde{T}_{kk})^{-1}(\tilde{A}_{kk} - \tilde{A}_{kk})$ of $\tilde{Z} = x(I - x\tilde{T})^{-1}(\tilde{A} - \tilde{T})$, where $\tilde{Z}_{kk}$ is assumed to vanish if the underlying eigenvalue of A is real ($k \notin K$). Hence, in order to have $\rho(Z(\infty)) = 0$, we choose the $\tilde{T}_{kk}$ with $k \in K$ such that the spectral radius of the corresponding diagonal blocks $\tilde{Z}_{kk}(x)$ vanishes at $x = \infty$.

We derive from (3.7) and (3.8) that the eigenvalues ζ_k of $\tilde{Z}_{kk}$ satisfy the characteristic equation

$$\det\begin{pmatrix} (a_k - u_k)x - \zeta_k(1 - xu_k) & b_k x \\ (c_k - v_k)x + \zeta_k v_k x & (2\xi_k - a_k - w_k)x - \zeta_k(1 - xw_k) \end{pmatrix} = 0. \tag{3.10}$$

It is easily verified that we always have one zero root if

$$v_k = b_k^{-1}(u_k - a_k)(2\xi_k - a_k - w_k) + c_k.$$

On substitution of b_k as defined in (3.7) we obtain the expression given in (3.9). Furthermore, the second root reads

$$\zeta_k(x) = x\frac{2\xi_k - u_k - w_k + x(u_k w_k - \alpha_k^2)}{(1 - xu_k)(1 - xw_k)}, \tag{3.11}$$

which vanishes at infinity if (3.9) is satisfied. This proves assertion (i).

Since u_k and w_k are positive for $\gamma_k > 0$, the matrix $\tilde{T}$ has positive eigenvalues, proving assertion (ii). The root $\zeta_k(x)$ assumes a maximal value at $x = -(u_k w_k)^{-1/2} = -\alpha_k^{-1}$ which is given by

$$\rho_k := 1 - \frac{2\gamma_k(\alpha_k + \xi_k)}{\alpha_k(\gamma_k + 1)^2}.$$

It is easily verified that ρ_k always satisfies $-1 < \rho_k < 1$, so that assertion (iii) follows.

In order to prove assertion (iv), we first show that integer powers of $Z(\infty)$ greater than 1 vanish. By observing that $Z^v = Q\tilde{Z}^v Q^{-1}$, we have to show that all positive integer powers of $\tilde{Z}(\infty)$ greater than 1 vanish. Evidently, if $\tilde{T}$ is block-diagonal, then $\tilde{Z}(z)$ is block-diagonal. Hence, $\tilde{Z}(\infty)$ is block-diagonal with diagonal blocks $\tilde{Z}_{kk}(\infty)$. By virtue of assertion (i), these blocks have a zero spectral radius, and consequently, $(\tilde{Z}_{kk}(\infty))^v$ vanishes for $v \geqslant 2$ (this can easily be verified by considering their Schur decompositions). This implies that $\tilde{Z}^v(\infty)$ itself, and hence $Z^v(\infty)$, vanishes for $v \geqslant 2$. It can be verified that

$$Z^v(x) = \sum_{i=1}^{\infty}(Z(\infty))^{[v/i]}O(x^{1-i}), \tag{3.12}$$

where for any real r, $[r]$ denotes the first integer greater than or equal to r. Hence, $Z^v(x) = O(x^{1-v})$ as $x \to \infty$. Substituting into (3.3) yields the fourth assertion of the theorem. □

From this theorem it follows that $\rho_{\max}$ is minimized if all γ_k equal 1. However, if $\gamma_k = 1$, then $u_k = w_k$, so that $\tilde{T}$, and hence B, is defective. This means that we cannot diagonalize the iteration process (2.2) into the form (2.3). Therefore, we shall choose γ_k close to but distinct from 1. For example, if all γ_k equal $\frac{7}{8}$, then u_k and w_k are well separated and

$$\rho_{\max} = \max\left\{\left|1 - \frac{112}{225}(\alpha_k + \xi_k)\alpha_k^{-1}\right| : k \in K\right\},$$

whereas the minimal value is given by $\rho_{\max} = \max\{|1 - \frac{1}{2}(\alpha_k + \xi_k)\alpha_k^{-1}|:\ k \in K\}$.

3.3.2. Choice of the free parameters

As already remarked, the strictly lower triangular blocks of $\tilde{T}$, and the parameters a_k and c_k are still free. We shall choose these strictly lower triangular blocks zero, so that according to Theorem 3.2, $\rho^{(\nu)}(x)$ vanishes at infinity for $\nu \geqslant 2$. The free parameters a_k and c_k can be used for reducing the magnitude of $\rho^{(1)} = \max\{\|Z(x)\|:\ x \leqslant 0\}$. One option is to minimize $\|Z(x)\|$ in the inequality $\|Z(x)\| \leqslant \kappa(Q)\|\tilde{Z}(x)\|, \kappa(Q)$ being the condition number of Q. This can be achieved by minimizing the values of $\|\tilde{Z}_{kk}(x)\|$. The representation

$$\tilde{Z}_{kk}(x) = \frac{x}{(1 - u_k x)(1 - w_k x)} \times \begin{pmatrix} (a_k - u_k)(1 - w_k x) & b_k(1 - w_k x) \\ (a_k - u_k)v_k x + (c_k - v_k)(1 - u_k x) & b_k v_k x + (2\xi_k - a_k - w_k)(1 - u_k x) \end{pmatrix},$$

suggests choosing $a_k = u_k$, to obtain

$$a_k = \gamma_k \alpha_k, \quad v_k = c_k, \quad w_k = \alpha_k/\gamma_k, \tag{3.13}$$

and

$$\tilde{Z}_{kk} = \begin{pmatrix} 0 & \dfrac{\theta_k}{c_k} \\ 0 & \zeta_k \end{pmatrix}, \quad \zeta_k(x) = x\frac{2\xi_k - a_k - w_k}{(1 - x a_k)(1 - x w_k)}, \quad \theta_k(x) := -\frac{(a_k^2 - 2\xi_k a_k + \alpha_k^2)x}{1 - a_k x}$$

with c_k still a free parameter. Since $\zeta_k(x)$ is a function with fixed coefficients, the maximum norm of $\tilde{Z}_{kk}$ is minimized if

$$|c_k| \geqslant \frac{\max\{|\theta_k(x)|:\ x \leqslant 0\}}{\max\{|\zeta_k(x)|:\ x \leqslant 0\}}. \tag{3.14}$$

From (3.13) and (3.14) we obtain the method

$$\tilde{T} := \begin{pmatrix} \tilde{T}_{11} & O & O & O & \cdots \\ O & \tilde{T}_{22} & O & O & \cdots \\ O & O & \tilde{T}_{33} & O & \cdots \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{pmatrix},$$

$$\tilde{T}_{kk} := \begin{pmatrix} \gamma_k \alpha_k & 0 \\ c_k & \dfrac{\alpha_k}{\gamma_k} \end{pmatrix} \text{ if } k \in K, \ \tilde{T}_{kk} = \xi_k \text{ otherwise,} \tag{3.15}$$

where

$$|c_k| \geqslant \frac{(\gamma_k + 1)^2}{\gamma_k} \alpha_k.$$

3.3.3. Construction of Q

For the methods generated by (3.15), there is still some freedom in choosing $\tilde{T}$ and Q. For each c_k, the matrix $\tilde{A}$ is fixed and defines a family of transformation matrices Q satisfying the relation $Q\tilde{A} = AQ$. This family can be generated by a procedure described in [8]. Within this family, we have determined the matrix for which $\rho^{(1)}$ is numerically minimized. In the case of the four-stage Radau IIA corrector we found for $\gamma_k = \frac{7}{8}$ the following numerically optimal method parameters:

$$\tilde{T} = \begin{pmatrix} 0.03448384 & 0 & 0 & 0 \\ -0.15834419 & 0.04504012 & 0 & 0 \\ 0 & 0 & 0.026431456 & 0 \\ 0 & 0 & -0.12136894 & 0.03452272 \end{pmatrix},$$

$$Q = \begin{pmatrix} 0.38205380 & 0.01709570 & -0.32651514 & -0.13054141 \\ 0.26713523 & -0.07242663 & 0.59303366 & 0.33355256 \\ 0.82772826 & -0.52316543 & 0.87439479 & -0.22432712 \\ -1.40177558 & -1.54184094 & -2.48244565 & -1.62324383 \end{pmatrix}, \qquad (3.16)$$

$$B = \begin{pmatrix} 0.00069709 & -0.02327295 & 0.01324386 & -0.00389225 \\ 0.09133373 & 0.09490827 & -0.03178816 & 0.00945629 \\ 0.11486891 & 0.03494592 & 0.06066531 & -0.00566972 \\ 0.09129004 & -0.07918010 & 0.19322700 & -0.01579253 \end{pmatrix},$$

with amplification factors

$$\rho = (7.93, 1.30, 1.06, 0.95, \ldots, 0.69).$$

Notice that the $\rho^{(\nu)}$ values for the PILSRKN method (3.5) are much better. On the other hand, for (3.16), the accumulated amplification matrix $Z^{\nu}(x)$ vanishes at infinity if $\nu \geqslant 2$, so that the stiff error components are more or less removed from the iteration error within two iterations, whereas it takes four iterations in the case of (3.5).

3.4. Orthogonal transformations

In order to have fast convergence right from the beginning, we should have small initial averaged amplification factors $\rho^{(\nu)}$. To achieve this it is not sufficient to have a small asymptotic amplification factor $\rho_{\max}$, but the condition number of the transformation matrix should also be sufficiently small. The most ideal case is to look for orthogonal transformation matrices Q. One obvious option for choosing a family of orthogonal matrices Q are the permutation matrices. By means of a suitable permutation, we may try to move the entries of large magnitude in the lower left corner of the transformed matrix. However, in the RKN correctors we have in mind (i.e. the classical Gauss–Legendre and Radau IIA correctors), the matrix A already has its larger entries in the lower left corner. An alternative family of orthogonal transformation matrices consists of rotation matrices:

$$Q = \mathrm{diag}(Q_{kk}), \quad Q_{kk} := \begin{pmatrix} \cos(\phi_k) & -\sin(\phi_k) \\ \sin(\phi_k) & \cos(\phi_k) \end{pmatrix} \text{ if } k \in K, \quad Q_{kk} = 1 \text{ if } k \notin K, \tag{3.17}$$

where the ϕ_k are free parameters. Such transformation matrices yield only a minor rearrangement of the magnitudes of the matrix entries. Given the matrix A and the parameters ϕ_k, we apply Theorem 3.1 by computing the Crout decomposition LU for the transformed RKN matrices $\tilde{A} := Q^{-1}AQ$, to obtain $\tilde{T}=L$ and $B = QLQ^{-1}$. Then, by evaluating the corresponding maximal amplification factor $\rho_{\max}$ and by minimizing $\rho_{\max}$ over the parameters ϕ_k, we find the matrices Q which are optimal in the class (3.17). This procedure was carried out for the 4-stage Radau IIA corrector:

$$\begin{gathered}\tilde{T} = \begin{pmatrix} 0.04467745 & 0 & 0 & 0 \\ 0.04236621 & 0.01258375 & 0 & 0 \\ 0.17376891 & 0.10910205 & 0.09118815 & 0 \\ 0.32687760 & 0.24513629 & 0.26054917 & 0.02764423 \end{pmatrix}, \\ Q = \begin{pmatrix} 0.68929086 & -0.72448472 & 0 & 0 \\ 0.72448472 & 0.68929086 & 0 & 0 \\ 0 & 0 & 0.99328690 & 0.11567681 \\ 0 & 0 & -0.11567681 & 0.99328690 \end{pmatrix}, \\ B = \begin{pmatrix} 0.00667530 & -0.00621012 & 0 & 0 \\ 0.03615609 & 0.05058590 & 0 & 0 \\ 0.04598076 & 0.24668626 & 0.12027503 & -0.01078765 \\ 0.04268388 & 0.37980180 & 0.24976152 & -0.00144265 \end{pmatrix},\end{gathered} \tag{3.18}$$

with amplification factors

$$\rho = (0.79, 0.75, 0.68, 0.65, \ldots, 0.61).$$

With respect to its $\rho^{(\nu)}$ values, the PILSRKN method defined by (3.18) is superior to (3.5) and to (3.16) as well.

4. Stability

In practice, the PILSRKN method will not be applied until convergence, so that the Newton iterates are not exactly computed. As a consequence, we do not get the corrector stability, that is, if A originates from a Gauss–Legendre method or Radau IIA method for first order IVPs, then we do not automatically get an A-stable or L-stable method for the second-order IVP (1.1). In order to derive the stability matrix we assume that each outer iteration consists of r inner iterations and that the predictor formula is of the type

$$Y_n^{(0,r)} = (P \otimes I) Y_{n-1}^{(m,r)}, \tag{4.1}$$

where P is an $s \times s$ matrix. If P is such that $Y_n^{(0,r)}$ has maximal order $q = s - 1$, then it will be called the extrapolation (EPL) predictor, and if $P = ee_s^T$ (with e_s denoting the sth unit vector), then it will be called the last step value (LSV) predictor. Let us define

$$G(\Delta) := F(Y_n + \Delta) - F(Y_n) - (I \otimes J)\Delta, \qquad N := (I - A \otimes h^2 J)^{-1}(A \otimes I). \tag{4.2}$$

Setting $Y_n^{(j,0)} := Y_n^{(j-1,r)}$, we find by a simple manipulation that

$$Y_n^{(j,r)} - Y_n = M^r(Y_n^{(j-1,r)} - Y_n) + h^2(I - M^r)NG(Y_n^{(j-1,r)} - Y_n), \quad j = 1, \dots, m, \tag{4.3}$$

where M is defined in (3.1). For the stability test equation $y'' = \lambda y$, we obtain

$$G(Y_n^{(j-1,r)} - Y_n) = \mathbf{0}, \qquad Y_n = (I - xA)^{-1}(y_{n-1} + hcy'_{n-1}), \quad x := h^2\lambda$$

so that

$$Y_n^{(m,r)} = (I - Z^{mr})(I - xA)^{-1}(ey_{n-1} + chy'_{n-1}) + Z^{mr}PY_{n-1}^{(m,r)}. \tag{4.4}$$

Similarly, the step point formulas (1.4) take the form

$$\begin{aligned} y_n - b^{\mathrm{T}}A^{-1}Y_n^{(m,r)} &= y_{n-1} + hy'_{n-1} - b^{\mathrm{T}}A^{-1}ey_{n-1} - b^{\mathrm{T}}A^{-1}chy'_{n-1}, \\ hy'_n - d^{\mathrm{T}}A^{-1}Y_n^{(m,r)} &= hy'_{n-1} - d^{T}A^{-1}ey_{n-1} - d^{T}A^{-1}chy'_{n-1}, \end{aligned} \tag{4.5}$$

to obtain the stability matrix

$$R(x) := \begin{pmatrix} I & 0 & 0 \\ -b^{\mathrm{T}}A^{-1} & 1 & 0 \\ -d^{\mathrm{T}}A^{-1} & 0 & 1 \end{pmatrix}^{-1}$$

$$\times \begin{pmatrix} Z^{mr}(x)P & (I - Z^{mr}(x))(I - xA)^{-1}e & (I - Z^{mr}(x))(I - xA)^{-1}c \\ \mathbf{0}^{\mathrm{T}} & 1 - b^{\mathrm{T}}A^{-1}e & 1 - b^{\mathrm{T}}A^{-1}c \\ \mathbf{0}^{\mathrm{T}} & -d^{\mathrm{T}}A^{-1}e & 1 - d^{\mathrm{T}}A^{-1}c \end{pmatrix} \tag{4.6}$$

(we remark that for Radau IIA correctors, we have $b^{\mathrm{T}}A^{-1} = e_s^T$). For the PC pairs (LSV, 4-stage Radau IIA) and (EPL, 4-stage Radau IIA), we found the stable mr-values as listed in Table 1. These figures clearly indicate that the LSV predictor yields a more stable overall process than the EPL predictor, particularly in the case of the Crout type and orthogonal Q type PILSRKN methods (3.5) and (3.18).

Table 1
Stable mr-values for 4-stage Radau IIA

Predictor	(3.5)	(3.16)	(3.18)
LSV	4	7	3
EPL	9	8	8

5. Numerical illustration

In this section, we illustrate the convergence behaviour when using the PILSRKN matrices (3.5), (3.16) and (3.18) for solving the Newton systems (2.1). In our experiments, we use the LSV predictor, the 4-stage Radau IIA corrector, the Shampine step point formulas (1.4), and constant stepsizes. In order to avoid round-off for small values of h in the iteration scheme and in the output formulas (1.4), we define the new variables

$$z_n := hy_n',$$

$$\boldsymbol{Z}^{(j,\nu)} := (S^{-1}\otimes I)(\boldsymbol{Y}_n^{(j,\nu)} - \boldsymbol{e}\otimes y_{n-1} - \boldsymbol{c}\otimes z_{n-1}) = \boldsymbol{X}^{(j,\nu)} - (S^{-1}\otimes I)(\boldsymbol{e}\otimes y_{n-1} + \boldsymbol{c}\otimes z_{n-1}),$$

where S is the diagonalizing matrix used in (2.3). Then, the method {(1.4), (2.1), (2.3)} can be implemented according to

```
$\boldsymbol{Z}^{(0,r)} = -(S^{-1}\otimes I)(\boldsymbol{c}\otimes z_{n-1})$,
for $j = 1$ to $m$
  $\boldsymbol{G}_n^{(j-1)} := h^2(S^{-1}A\otimes I)\boldsymbol{F}((S\otimes I)\boldsymbol{Z}^{(j-1,r)} + \boldsymbol{e}\otimes y_{n-1} + \boldsymbol{c}\otimes z_{n-1}) - (S^{-1}AS\otimes h^2J)\boldsymbol{Z}^{(j-1,r)}$
  for $\nu = 1$ to $r$
    $\boldsymbol{Z}^{(j,0)} = \boldsymbol{Z}^{(j-1,r)}$
    solve $(I - \tilde{B}\otimes h^2J)(\boldsymbol{Z}^{(j,\nu)} - \boldsymbol{Z}^{(j,\nu-1)}) = -(I - S^{-1}AS\otimes h^2J)\boldsymbol{Z}^{(j,\nu-1)} + \boldsymbol{G}_n^{(j-1)}$
$y_n = y_{n-1} + z_{n-1} + (\boldsymbol{b}^{\mathrm{T}}A^{-1}S\otimes I)\boldsymbol{Z}^{(m,r)}$
$z_n = z_{n-1} + (\boldsymbol{d}^{\mathrm{T}}A^{-1}S\otimes I)\boldsymbol{Z}^{(m,r)}$.
```

5.1. Iteration strategy

Our first concern is to get insight how the performance of the iteration process depends on the number of inner and outer iterations r and m. We illustrate this by means of the nonlinear orbit equation of Fehlberg (cf. [2]):

$$y''(t) = Jy(t), \qquad J := \begin{pmatrix} -4t^2 & -\frac{2}{r(t)} \\ \frac{2}{r(t)} & -4t^2 \end{pmatrix}, \qquad r(t) := \|y(t)\|_2; \qquad \sqrt{\pi/2} \leqslant t \leqslant 12\pi, \tag{5.1}$$

with exact solution $y(t) = (\cos(t^2), \sin(t^2))^{\mathrm{T}}$. We performed the iteration strategy test for the orthogonal Q type PILSRKN method generated by (3.18), because this method yields the most stable integration process. The Tables 2(a)–(d) present the minimal number of significant digits Δ of the components of y at the end point of the integration interval, that is, at the end point, the absolute errors are written as $10^{-\Delta}$ (negative Δ-values are indicated with $*$). Our first conclusion from these tables is that for solving the corrector equation, we need at least two outer iterations (i.e. $m \geqslant 2$). As soon as we impose this condition, there is hardly no difference between the accuracies obtained for constant values of mr. Because for given LU-decompositions of the diagonal blocks of the matrix $I - \tilde{B}\otimes h^2J$, the value of mr may be considered as a measure of the computational costs per step, our second conclusion is that we may perform a constant number of inner iterations.

Table 2(a)
Fehlberg problem, $h = 0.0228$

m	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$	$r = 6$
1	*	*	*	0.4	1.9	1.1
2	*	0.3	1.6	2.0	2.1	2.1
3	*	1.6	2.1	2.1		
4	0.3	2.0	2.1			
5	1.0	2.1				
6	1.6	2.1				

Table 2(b)
Fehlberg problem, $h = 0.0114$

m	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$	$r = 6$
1	*	*	1.2	2.2	2.0	2.0
2	*	2.4	4.1	4.2	4.2	4.2
3	1.1	4.1	4.2			
4	2.4	4.2				
5	3.6	4.2				
6	4.1	4.2				

Table 2(c)
Fehlberg problem, $h = 0.0057$

m	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$	$r = 6$
1	*	1.0	3.9	2.9	2.8	2.8
2	1.0	4.7	6.4	6.3	6.3	6.3
3	2.8	6.3	6.3			
4	4.7	6.3				
5	6.2	6.3				
6	6.3					

Table 2(d)
Fehlberg problem, $h = 0.00285$

m	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$	$r = 6$
1	*	2.1	3.8	3.7	3.7	3.7
2	2.1	7.1	8.4	8.4	8.4	8.4
3	4.6	8.4				
4	7.0	8.4				
5	8.4					
6	8.4					

5.2. Comparison of PILSRKN methods

In this section we compare the performance of the PILSRKN methods (3.5), (3.16) and (3.18). These comparisons were carried out for the Fehlberg problem (5.1), the Kramarz problem [9]

$$y''(t) = \begin{pmatrix} 2498 & 4998 \\ -2499 & -4999 \end{pmatrix} y(t), \qquad y(0) = \begin{pmatrix} 2 \\ -1 \end{pmatrix}, \qquad y'(0) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad 0 \leqslant t \leqslant 100, \quad (5.2)$$

with exact solution $y(t) = (2\cos(t), -\cos(t))^{\mathrm{T}}$, the Strehmel–Weiner problem [13]

$$\begin{aligned} y_1''(t) &= (y_1(t) - y_2(t))^3 + 6368y_1(t) - 6384y_2(t) + 42\cos(10t), \\ y_2''(t) &= -(y_1(t) - y_2(t))^3 + 12768y_1(t) - 12784y_2(t) + 42\cos(10t), \\ y(0) &= \tfrac{1}{2}\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad y'(0) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad 0 \leqslant t \leqslant 10, \end{aligned} \qquad (5.3)$$

with exact solution $y_1(t) = y_2(t) = \cos(4t) - \frac{1}{2}\cos(10t)$, and the Pleiades problem PLEI given in [4, p. 237]. The PLEI problem consists of 14 nonlinear orbit equations on the interval [0, 3].

We used one inner iteration ($r = 1$) and, in order to enable a mutual comparison, we chose the number of outer iterations one less than needed to really solve the corrector Eq. (1.3).

The results listed in the Tables 3–6 indicate that the method {(2.1), (3.16)} produces the highest accuracies if it converges. However, it is less robust than the methods {(2.1), (3.5)} and {(2.1), (3.18)} due to the development of instabilities (see also Table 1). Since {(2.1), (3.18)} is in almost all cases (slightly) more accurate than {(2.1), (3.5)}, our conclusion is that {(2.1), (3.18)} is the most attractive one of the three methods constructed in this paper.

Finally, we compare the efficiency of the methods of this paper with the diagonally implicit RKN method based on the 4-stage Radau IIA formula as developed in [11]. This method requires 5 sequential, singly diagonal-implicit stages per step. Effectively (on 4 processors), this is comparable

Table 3
Fehlberg problem, $m = 5$, $r = 1$

h	{(2.1),(3.5)}	{(2.1),(3.16)}	{(2.1),(3.18)}
0.0228	0.7	2.5	1.0
0.0114	3.3	4.2	3.6
0.0057	6.0	6.3	6.2
0.00285	8.3	8.4	8.4

Table 4
Kramarz problem, $m = 4$, $r = 1$

h	{(2.1),(3.5)}	{(2.1),(3.16)}	{(2.1),(3.18)}
0.8	2.5	4.1	2.8
0.4	4.9	6.9	5.2
0.2	7.3	*	7.6
0.1	9.7	*	10.0

Table 5
Strehmel–Weiner problem, $m = 5$, $r = 1$

h	$\{(2.1),(3.5)\}$	$\{(2.1),(3.16)\}$	$\{(2.1),(3.18)\}$
0.5	1.1	2.1	1.4
0.25	3.4	5.1	3.8
0.125	6.2	7.4	6.6
0.0625	9.1	9.9	9.4
0.03125	11.5	11.5	11.5

Table 6
PLEI problem from [4], $m = 4$, $r = 1$

h	$\{(2.1),(3.5)\}$	$\{(2.1),(3.16)\}$	$\{(2.1),(3.18)\}$
0.002	0.4	2.0	0.9
0.001	3.4	4.3	3.7
0.0005	5.9	6.2	6.0
0.00025	8.2	8.3	8.3
0.000125	10.4	10.3	10.3

with the computational needed in our methods when applied with $mr = 5$. For the Kramarz problem, [11, Table 6] reports for stepsizes $h=0.2$ and $h=0.1$ accuracies of 5.4 and 8.1 significant digits. From Table 4 it follows that {(2.1), (3.5)} and {(2.1), (3.18)} produce considerable higher accuracies for less computational effort ($mr = 4$, same stepsizes). Similarly, for the Strehmel–Weiner problem, [11, Table 10] reports for stepsizes $h = 0.05$ and $h = 0.025$ accuracies of 6.4 and 9.0 significant digits, whereas Table 5 again shows considerable higher accuracies for less computational effort ($mr = 5$, larger stepsizes).

References

[1] J.C. Butcher, On the implementation of implicit Runge–Kutta methods, BIT 16 (1976) 237–240.

[2] E. Fehlberg, Classical Runge–Kutta–Nyström formulas with stepsize control for differential equations of the form $x''=f(t,x)$ (German), Computing 10 (1972) 305–315.

[3] E. Hairer, Unconditionally stable methods for second order differential equations, Numer. Math. 32 (1979) 373–379.

[4] E. Hairer, S.P. Nørsett, G. Wanner, Solving Ordinary Differential Equations, I. Nonstiff Problems, Springer, Berlin, 1987.

[5] E. Hairer, G. Wanner, Solving Ordinary Differential Equations, II. Stiff and Differential-algebraic Problems, Springer, Berlin, 1991.

[6] P.J. van der Houwen, B.P. Sommeijer, Iterated Runge–Kutta methods on Parallel Computers, SIAM J. Sci. Statist. Comput. 12 (1991) 1000–1028.

[7] P.J. van der Houwen, J.J.B. de Swart, Triangularly Implicit Iteration Methods for ODE-IVP Solvers, SIAM. J. Sci. Comput. (1996), SIAM J. Sci. Comput. 18 (1997) 41–55.

[8] P.J. van der Houwen, J.J.B. de Swart, Parallel linear solvers for Runge–Kutta methods, Adv. Comput. Math. (1996), Adv. Comp. Math. 7 (1997) 157–181.

[9] L. Kramarz, Stability of collocation methods for the numerical solution of $y'' = f(x,y)$, BIT 20 (1980) 215–222.

[10] E. Messina, Convergence and stability plots for parallel linear solvers for use in Runge–Kutta–Nyström methods, 1996, in preparation.

[11] H.C. Nguyen, *A*-stable diagonally implicit Runge–Kutta–Nyström methods for parallel computers, Numer. Algorithms 4 (1993) 263–281.

[12] L.F. Shampine, Implementation of implicit formulas for the solution of ODEs, SIAM J. Sci. Statist. Comput. 1 (1980) 103–118.

[13] K. Strehmel, R. Weiner, Nonlinear stability and phase analysis for adaptive Nyström–Runge–Kutta methods (German), Computing 35 (1985) 325–344.