PERGAMON Computers and Mathematics with Applications 38 (1999) 17–30

An International Journal
computers & mathematics
with applications

www.elsevier.nl/locate/camwa

A General Class of Explicit Pseudo Two-Step RKN Methods on Parallel Computers

NGUYEN HUU CONG*
Faculty of Mathematics, Mechanics and Informatics
Hanoi University of Sciences
334 Nguyen Trai, Thanh Xuan, Hanoi, Vietnam

K. STREHMEL AND R. WEINER
FB Mathematik und Informatik, Martin-Luther-Universität Halle-Wittenberg
Theodor-Lieser-Str. 5, D-06120 Halle, Germany

(Received July 1998; revised and accepted May 1999)

Abstract—The aim of this paper is to investigate a general class of explicit pseudo two-step Runge-Kutta-Nyström methods (RKN methods) of arbitrarily high order for nonstiff problems for systems of special second-order differential equations $\mathbf{y}''(t) = \mathbf{f}(\mathbf{y}(t))$. Order and stability considerations show that we can obtain for any given p, a stable p^{th}-order explicit pseudo two-step RKN method requiring $p-2$ right-hand side evaluations per step of which each evaluation can be obtained in parallel. Consequently, on a multiprocessor computer, only one sequential right-hand side evaluation per step is required. By a few widely-used test problems, we show the superiority of the methods considered in this paper over both sequential and parallel methods available in the literature.

Keywords—Runge-Kutta-Nyström methods, Two-Step Runge-Kutta-Nyström methods, Stability, Parallelism.

1. INTRODUCTION

Consider the numerical solution of a nonstiff Initial Value Problem (IVP) for the system of special second-order Ordinary Differential Equations (ODEs)

$$\mathbf{y}''(t) = \mathbf{f}(\mathbf{y}(t)), \qquad \mathbf{y}(t_0) = \mathbf{y}_0, \qquad \mathbf{y}'(t_0) = \mathbf{y}'_0, \qquad t \in [t_0, T], \tag{1.1}$$

where $\mathbf{y}, \mathbf{f} \in \mathbb{R}^d$. There are many important problems in celestial mechanics which are of form (1.1). The most efficient numerical methods for solving this problem are the explicit Runge-Kutta-Nyström-type methods (RKN-type methods). In the literature, sequential explicit RKN methods up to order ten can be found in, e.g., [1–5]. In order to exploit the facilities of parallel computers, several classes of parallel RKN-type methods have been investigated in, e.g., [6–11]. A common

This work was partly supported by DAAD, NRPFS, and Halle University.
*Author to whom all correspondance should be addressed.

0898-1221/99/$ - see front matter. Typeset by AMS-TEX
PII:S0898-1221(99)00211-4

challenge in the latter-mentioned papers is to reduce, for a given order of accuracy, the required number of sequential f-evaluations per step, using parallel processors. In the present paper, we investigate a general class of Explicit Pseudo Two-step RKN methods (EPTRKN methods) for the numerical solution of problem (1.1). The two-step nature of the methods considered in this paper is similar to that of the methods investigated in [12,13] for first-order ODEs and in [9,14] for second-order ODEs. Specifically, these EPTRKN methods are similar to the explicit pseudo two-step RK methods considered in [15] but the EPTRKN methods are defined more simply. Our approach in this paper is to approximate the stage values of an implicit RKN method at the present step by an explicit formula using the stage values from the preceding step. In this way, we can obtain an s-stage EPTRKN method possessing order up to $s+2$ and stage order up to $s+1$, requiring s f-evaluations per step (see Section 2.1). However, each of these s f-evaluations can be obtained in parallel. Consequently, when an s-stage EPTRKN method is implemented on an s-processor computer, only *one* sequential f-evaluation per step is required. This cheap computational cost leads to increased efficiency of EPTRKN methods when compared to the parallel and sequential methods from the literature (see Section 3). Stability considerations reveal that for any EPTRKN method, the zero-stability requirement is verified. The stability regions of a number of specified EPTRKN methods used in the numerical experiments are effectively large for nonstiff problems (cf. Section 2.2 and Section 3).

In the following sections, for the sake of simplicity of notation, we assume that the IVP (1.1) is a scalar problem. However, all considerations below can be straightforwardly extended to a system of ODEs, and therefore, also to nonautonomous equations.

2. EXPLICIT PSEUDO TWO-STEP RKN METHODS

The definition of explicit pseudo two-step RKN methods (EPTRKN methods) can be directly worked out using collocation techniques as in [15] for first-order ODEs. In this paper, we apply a more simple approach by starting with an s-stage collocation Implicit RKN method (IRKN method) based on s-dimensional collocation vector $\mathbf{c} = (c_1, \ldots, c_s)^\top$. This IRKN method will be referred to as the *generating IRKN method*. For a scalar and autonomous problem of form (1.1), the generating IRKN method reads

$$\mathbf{U}_n = u_n\mathbf{e} + hu'_n\mathbf{c} + h^2 A_N f(\mathbf{U}_n), \tag{2.1a}$$

$$u_{n+1} = u_n + hu'_n + h^2\mathbf{b}^\top f(\mathbf{U}_n), \tag{2.1b}$$

$$u'_{n+1} = u'_n + h\mathbf{d}^\top f(\mathbf{U}_n). \tag{2.1c}$$

Here, $u_{n+1} \approx y(t_{n+1})$, $u'_{n+1} \approx y'(t_{n+1})$, $\mathbf{U}_n \approx y(t_n\mathbf{e}+\mathbf{c}h) = [y(t_n+c_1h), \ldots, y(t_n+c_sh)]^\top$ at n^{th} step, h is the stepsize, the $s \times s$ matrix A_N, and the s-dimensional vectors $\mathbf{b}$ and $\mathbf{d}$ are the RKN parameters, $\mathbf{e}$ is the s-dimensional vector with unit entries. Furthermore, in (2.1) and elsewhere in this paper, we use for any vector $\mathbf{v} = (v_1, \ldots, v_s)^\top$ and any scalar function f the notation $f(\mathbf{v}) := [f(v_1), \ldots, f(v_s)]^\top$.

Replacing the implicit stage vector equation (2.1a) in (2.1) by an explicit approximation formula using the stage vector from the preceding step leads us to the method

$$\mathbf{Y}_n = y_n\mathbf{e} + hy'_n\mathbf{c} + h^2 A f(\mathbf{Y}_{n-1}), \tag{2.2a}$$

$$y_{n+1} = y_n + hy'_n + h^2\mathbf{b}^\top f(\mathbf{Y}_n), \tag{2.2b}$$

$$y'_{n+1} = y'_n + h\mathbf{d}^\top f(\mathbf{Y}_n). \tag{2.2c}$$

Here as in (2.1), $y_{n+1} \approx y(t_{n+1})$, $y'_{n+1} \approx y'(t_{n+1})$, $\mathbf{Y}_n \approx y(t_n\mathbf{e} + \mathbf{c}h)$. The method parameter matrix A will be determined by order conditions (see Section 2.1). Method (2.2) slightly differs from its generating IRKN method by the explicit relation (2.2a). This method is similar to a RKN method but it is not a RKN method nor a two-step RKN method of [9,14]. It has no implicit

relation and carries the information from the preceding step, and therefore, will be termed s-stage explicit pseudo two-step Runge-Kutta-Nyström method (EPTRKN method) based on collocation vector $\mathbf{c}$. For convenience of presentation, we specify this EPTRKN method by the tableau:

A	$\mathbf{c}$	O
	y_{n+1}	$\mathbf{b}^{\mathsf{T}}$
	y'_{n+1}	$\mathbf{d}^{\mathsf{T}}$

(2.3)

In order to start method (2.2), an appropriate starting procedure is needed to generate a sufficiently accurate starting vector $\mathbf{Y}_0$ from y_0 and y'_0. This can be done, for example, by using an appropriate PIRKN-type or BPIRKN-type method (cf., e.g., [7,11,16]).

For the EPTRKN method (2.2), at each step, we need to compute $2s$ f-evaluations of two s-dimensional vectors $f(\mathbf{Y}_{n-1})$ and $f(\mathbf{Y}_n)$. However, s f-evaluations of $f(\mathbf{Y}_{n-1})$ are already available from the preceding step so that only s f-evaluations of $f(\mathbf{Y}_n)$ are required. These s f-evaluations can be evaluated in parallel on s processors. Consequently, the s-stage EPTRKN method (2.2) implemented on an s-processor computer, requires just *one* sequential f-evaluation per step.

The order of EPTRKN method (2.2) can be studied in the same way as the order of a RKN method. Thus suppose that $y_n = y(t_n)$ and $\mathbf{Y}_{n-1} = y(t_{n-1}\mathbf{e} + h\mathbf{c})$. Then we have the following definition.

DEFINITION 2.1. *The EPTRKN method (2.2) will be of order p^* if*

$$y(t_{n+1}) - y_{n+1} = O\left(h^{p^*+1}\right),$$
$$y'(t_{n+1}) - y'_{n+1} = O\left(h^{p^*+1}\right),$$

and stage order $q^* = \min\{p^*, q\}$, if in addition,

$$y\left(t_n\mathbf{e} + h\mathbf{c}\right) - \mathbf{Y}_n = O\left(h^{q+1}\right).$$

Notice that the local stage order equals $q+1$. Now we shall consider the order conditions for the explicit approximation formula (2.2a).

2.1. Order Conditions

For the (fixed) stepsize h, the q^{th}-order conditions for (2.2a) can be derived by replacing $\mathbf{Y}_{n-1}$, y_n, and $\mathbf{Y}_n$ by the exact solution values $y(t_{n-1}\mathbf{e}+h\mathbf{c}) = y(t_n\mathbf{e}+h(\mathbf{c}-\mathbf{e}))$, $y(t_n)$, and $y(t_n\mathbf{e}+h\mathbf{c})$, respectively. On substitution of these exact solution values into (2.1), we are led to the relation

$$y(t_n\mathbf{e} + h\mathbf{c}) - y(t_n)\mathbf{e} - hy'(t_n)\mathbf{c} - h^2Ay''(t_n\mathbf{e} + h(\mathbf{c} - \mathbf{e})) = O\left(h^{q+1}\right). \tag{2.4}$$

Using Taylor expansions in the neighbourhood of t_n, we can expand the left-hand side of (2.4) in powers of h and obtain

$$\begin{aligned}&\left[\exp\left(h\mathbf{c}\frac{d}{dt}\right) - \mathbf{e} - h\mathbf{c}\frac{d}{dt} - h^2A\frac{d^2}{dt^2}\exp\left(h(\mathbf{c}-\mathbf{e})\frac{d}{dt}\right)\right]y(t_n)\\ &= \sum_{j=1}^{q-1}\mathbf{C}^{(j+1)}\left(h\frac{d}{dt}\right)^{j+1}y(t_n) + \mathbf{C}^{(q+1)}\left(h\frac{d}{dt}\right)^{q+1}y(t^*) = O\left(h^{q+1}\right),\end{aligned} \tag{2.5}$$

where, t^* is a suitably chosen point in the interval $[t_{n-1}, t_{n+1}]$, and

$$\mathbf{C}^{(j+1)} = \frac{1}{(j+1)!}\left[\mathbf{c}^{j+1} - j(j+1)A(\mathbf{c}-\mathbf{e})^{j-1}\right], \qquad j = 1, \ldots, q-1. \tag{2.6a}$$

The vectors $\mathbf{C}^{(j+1)}$ represent the error vectors of the numerical stage approximations. From (2.5), we obtain the order conditions

$$\mathbf{C}^{(j+1)} = \mathbf{0}, \qquad j = 1, \ldots, q-1. \tag{2.6b}$$

The vector $\mathbf{C}^{(q+1)}$ is the *principal* error vector of the stage numerical approximations in (2.2). Conditions (2.6) imply the following local order relation:

$$y(t_n\mathbf{e} + h\mathbf{c}) - \mathbf{Y}_n = O\left(h^{q+1}\right). \tag{2.7}$$

THEOREM 2.1. *Let p be the order of the generating IRKN method (2.1). If the function f is Lipschitz continuous, and if*

$$A(\mathbf{c} - \mathbf{e})^{j-1} = \frac{\mathbf{c}^{j+1}}{j(j+1)}, \qquad j = 1, \ldots, q-1, \tag{2.8}$$

then the EPTRKN method (2.2) has order $p^* = \min\{p, q+1\}$ and stage order $q^* = \min\{p^*, q\}$ for any collocation vector $\mathbf{c}$ with distinct abscissas c_i.

PROOF. Suppose that $y_n = y(t_n)$ and $\mathbf{Y}_{n-1} = y(t_{n-1}\mathbf{e} + h\mathbf{c})$. Since conditions (2.6) and (2.8) are equivalent, f is Lipschitz continuous, and from (2.7), we obtain the following order relations:

$$\begin{aligned} y(t_{n+1}) - y_{n+1} &= y(t_{n+1}) - y(t_n) - hy'(t_n) - h^2\mathbf{b}^\top f\left(y\left(t_n\mathbf{e} + h\mathbf{c}\right)\right) \\ &\quad + h^2\mathbf{b}^\top\left[f\left(y\left(t_n\mathbf{e} + h\mathbf{c}\right)\right) - f\left(\mathbf{Y}_n\right)\right] \\ &= O\left(h^{p+1}\right) + O\left(h^2\right)\left[y\left(t_n\mathbf{e} + h\mathbf{c}\right) - \mathbf{Y}_n\right] = O\left(h^{p+1}\right) + O\left(h^{q+3}\right), \end{aligned} \tag{2.9a}$$

$$\begin{aligned} y'(t_{n+1}) - y'_{n+1} &= y'(t_{n+1}) - y'(t_n) - h\mathbf{d}^\top f\left(y\left(t_n\mathbf{e} + h\mathbf{c}\right)\right) \\ &\quad + h\mathbf{d}^\top\left[f\left(y\left(t_n\mathbf{e} + h\mathbf{c}\right)\right) - f\left(\mathbf{Y}_n\right)\right] \\ &= O\left(h^{p+1}\right) + O(h)\left[y\left(t_n\mathbf{e} + h\mathbf{c}\right) - \mathbf{Y}_n\right] = O\left(h^{p+1}\right) + O\left(h^{q+2}\right). \end{aligned} \tag{2.9b}$$

In view of the order relations (2.9) Theorem 2.1 is proved. ∎

In order to express the parameter matrix A explicitly in terms of the collocation vector $\mathbf{c}$, we define the following matrices:

$$P := \left(\frac{\mathbf{c}^2}{2}, \frac{\mathbf{c}^3}{3}, \frac{\mathbf{c}^4}{4}, \ldots, \frac{\mathbf{c}^{s+1}}{s+1}\right), \qquad Q := \left(\mathbf{e}, 2(\mathbf{c} - \mathbf{e}), 3(\mathbf{c} - \mathbf{e})^2, \ldots, s(\mathbf{c} - \mathbf{e})^{s-1}\right).$$

Then the order conditions (2.6) can be presented in the form

$$AQ - P = O. \tag{2.10}$$

Since the components c_i of the vector $\mathbf{c}$ are assumed to be distinct, the matrix Q is nonsingular, and from (2.10), we may write

$$A = PQ^{-1}. \tag{2.11}$$

From condition (2.11) it follows that

$$y(t_n\mathbf{e} + h\mathbf{c}) - \mathbf{Y}_n = O\left(h^{s+2}\right). \tag{2.12}$$

Moreover, the order investigations for collocation IRKN methods ensure that the order p of the generating IRKN method (2.1) is at least equal to s (see, e.g., [17]). By a special choice of the collocation vector $\mathbf{c}$, it is possible to increase the order p beyond s (superconvergence) by satisfying the orthogonality relation (see [18, p. 212]), that is if the orthogonality relation

$$P_j(1) = 0, \qquad P_j(x) := \int_0^x \xi^{j-1} \prod_{i=1}^{s} (\xi - c_i)\, d\xi, \qquad j = 1, 2, \ldots, l \tag{2.13}$$

is satisfied then $p = s + l$. The following theorem is a consequence of Theorem 2.1 and the order relation (2.12).

THEOREM 2.2. *An s-stage EPTRKN method defined by (2.2) has order $p^* = s$ and stage order $q^* = s$ for any set of distinct collocation points c_i. It has stage order $q^* = s+1$ and order $p^* = s+1$ or $s+2$ if in addition, the orthogonality relation (2.13) is satisfied for $l = 1$ or $l \geq 2$, respectively.*

Thus, an s-stage EPTRKN method can attain the order $p^* = s + 2$.

According to the analysis of the local errors in this section, the starting values $\mathbf{Y}_0$ should be of local order $s+2$ that is

$$y(t_0\mathbf{e} + h\mathbf{c}) - \mathbf{Y}_0 = O\left(h^{s+2}\right).$$

In the following applications of EPTRKN methods, in the first step, the starting values of $\mathbf{Y}_0$ and y_1 of an EPTRKN method will be always generated by the associated PIRKN method using the collocation RKN corrector based on the same collocation vector $\mathbf{c}$ as the underlying EPTRKN method.

REMARK 2.1. For determining the parameters of an EPTRKN method, the matrix A can be defined by using (2.11), the vectors $\mathbf{b}$ and $\mathbf{d}$ are defined as the corresponding parameters of the generating direct or indirect collocation IRKN methods (see [17]).

REMARK 2.2. The method parameter matrix A defined by (2.11) can be used only for fixed stepsize implementations. With a variable stepsize strategy, the matrix A has to be recalculated when the stepsize changes. The entries of the variable matrix A are then, a function of the ratios of stepsizes. Variable stepsize strategy EPTRKN methods are the subject of [19].

A NUMERICAL TEST. We will test the theoretical order behaviour considered above of EPTRKN methods with some methods from a family of two-stage EPTRKN methods generated by the direct collocation IRKN method based on the vector $\mathbf{c} = (c_1, c_2)^\top$. This family is defined by the following tableau (cf. [17], (2.3), (2.11)).

$\frac{c_1^2(3c_2 - c_1 - 3)}{6(c_2 - c_1)}$	$\frac{c_1^2(3 - 2c_1)}{6(c_2 - c_1)}$	c_1	0	0
$\frac{c_2^2(2c_2 - 3)}{6(c_2 - c_1)}$	$\frac{c_2^2(c_2 - 3c_1 + 3)}{6(c_2 - c_1)}$	c_2	0	0
		y_{n+1}	$\frac{3c_2 - 1}{6(c_2 - c_1)}$	$\frac{1 - 3c_1}{6(c_2 - c_1)}$
		y'_{n+1}	$\frac{2c_2 - 1}{2(c_2 - c_1)}$	$\frac{1 - 2c_1}{2(c_2 - c_1)}$

(2.14)

We consider the EPTRKN methods based on the following collocation vectors $\mathbf{c}$:

$$\left(\frac{1}{2}, 1\right)^\top, \quad \left(\frac{1}{3}, 1\right)^\top, \quad \left(0, \frac{2}{3}\right)^\top, \quad \left(\frac{3 - \sqrt{3}}{6}, \frac{3 + \sqrt{3}}{6}\right)^\top.$$

By a direct calculation using (2.14), we obtain the parameters of the resulting EPTRKN methods which are specified by the following tableaux.

$-\frac{1}{24}$	$\frac{1}{6}$	$\frac{1}{2}$	0	0
$-\frac{1}{3}$	$\frac{5}{6}$	1	0	0
		y_{n+1}	$\frac{2}{3}$	$-\frac{1}{6}$
		y'_{n+1}	1	0

(2.15a)

$$\begin{array}{cc|c|cc} -\frac{1}{108} & \frac{7}{108} & \frac{1}{3} & 0 & 0 \\ -\frac{1}{3} & \frac{5}{6} & 1 & 0 & 0 \\ \hline & & y_{n+1} & \frac{1}{2} & 0 \\ & & y'_{n+1} & \frac{3}{4} & \frac{1}{4} \end{array} \tag{2.15b}$$

$$\begin{array}{cc|c|cc} 0 & 0 & 0 & 0 & 0 \\ -\frac{5}{27} & \frac{11}{27} & \frac{2}{3} & 0 & 0 \\ \hline & & y_{n+1} & \frac{1}{4} & \frac{1}{4} \\ & & y'_{n+1} & \frac{1}{4} & \frac{3}{4} \end{array} \tag{2.15c}$$

$$\begin{array}{cc|c|cc} \frac{5-3\sqrt{3}}{18} & \frac{3\sqrt{3}-4}{36} & \frac{3-\sqrt{3}}{6} & 0 & 0 \\ -\frac{4+3\sqrt{3}}{36} & \frac{5+3\sqrt{3}}{18} & \frac{3+\sqrt{3}}{6} & 0 & 0 \\ \hline & & y_{n+1} & \frac{3+\sqrt{3}}{12} & \frac{3-\sqrt{3}}{12} \\ & & y'_{n+1} & \frac{1}{2} & \frac{1}{2} \end{array} \tag{2.15d}$$

From Theorem 2.2 and the theoretical orders of the generating IRKN methods, method (2.15a) has $p^* = q^* = 2$; methods (2.15b) and (2.15c) have $p^* = q^* = 3$; and the last method (2.15d) based on Gauss-Legendre collocation vector has $p^* = 4$, $q^* = 3$.

We now apply these specified EPTRKN methods (2.15) to the linear nonautonomous problem (cf., e.g., [7–9])

$$\begin{aligned} \frac{d^2\mathbf{y}(t)}{dt^2} &= \begin{pmatrix} -2\alpha(t)+1 & -\alpha(t)+1 \\ 2(\alpha(t)-1) & \alpha(t)-2 \end{pmatrix} \mathbf{y}(t), \\ \alpha(t) &= \max\left\{2\cos^2(t), \sin^2(t)\right\}, \qquad 0 \le t \le 20, \\ \mathbf{y}(0) &= (0,0)^{\mathsf{T}}, \qquad \mathbf{y}'(0) = (-1,2)^{\mathsf{T}}, \end{aligned} \tag{2.16}$$

with exact solution $\mathbf{y}(t) = (-\sin(t), 2\sin(t))^{\mathsf{T}}$. The values of the Number of Correct Decimal digits NCD obtained at the end point of the integration interval for a given total number of integration steps N_{stp} (cf., e.g., [7–9]) listed in Table 1 nicely show the theoretical order behaviour of the EPTRKN methods as was stated in Section 2.1 (cf. Theorem 2.2).

Table 1. NCD-values at $t = 10$ for problem (3.1) obtained by p^{th}-order EPTRKN methods.

Methods	p	$N_{\text{stp}} = 1600$	$N_{\text{stp}} = 3200$	$N_{\text{stp}} = 6400$	$N_{\text{stp}} = 12800$	$N_{\text{stp}} = 25600$
$\text{EPTRKN}_{(2.15a)}$	2	3.7	4.3	4.9	5.5	6.1
$\text{EPTRKN}_{(2.15b)}$	3	5.6	6.5	7.4	8.3	9.2
$\text{EPTRKN}_{(2.15c)}$	3	5.6	6.5	7.4	8.3	9.2
$\text{EPTRKN}_{(2.15d)}$	4	6.9	8.1	9.3	10.5	11.7

2.2. Stability

Since EPTRKN method (2.2) is of two-step nature, we first check its property of zero-stability and next define its stability region. For that purpose we apply (2.2) to the model test equation $y''(t) = \lambda y(t)$, where λ runs through the eigenvalues of the Jacobian matrix $\frac{\partial \mathbf{f}}{\partial \mathbf{y}}$ which are assumed to be negative real. For this model test equation, the EPTRKN method (2.2) assumes the form

$$\mathbf{Y}_n = y_n\mathbf{e} + hy'_n\mathbf{c} + xA\mathbf{Y}_{n-1}, \tag{2.17a}$$

$$\begin{aligned} y_{n+1} &= y_n + hy'_n + x\mathbf{b}^\top\mathbf{Y}_n \\ &= y_n + hy'_n + x\mathbf{b}^\top(y_n\mathbf{e} + hy'_n\mathbf{c} + xA\mathbf{Y}_{n-1}) \\ &= \left(1 + x\mathbf{b}^\top\mathbf{e}\right) y_n + \left(1 + x\mathbf{b}^\top\mathbf{c}\right) hy'_n + x^2\mathbf{b}^\top A\mathbf{Y}_{n-1}, \end{aligned} \tag{2.17b}$$

$$\begin{aligned} hy'_{n+1} &= hy'_n + x\mathbf{d}^\top\mathbf{Y}_n \\ &= hy'_n + z\mathbf{d}^\top(y_n\mathbf{e} + hy'_n\mathbf{c} + xA\mathbf{Y}_{n-1}) \\ &= x\mathbf{d}^\top\mathbf{e}y_n + \left(1 + x\mathbf{d}^\top\mathbf{c}\right) hy'_n + x^2\mathbf{d}^\top A\mathbf{Y}_{n-1}, \end{aligned} \tag{2.17c}$$

where $x = \lambda h^2$. Relations (2.17) lead us to the recursion

$$\begin{pmatrix} \mathbf{Y}_n \\ y_{n+1} \\ hy'_{n+1} \end{pmatrix} = M(x) \begin{pmatrix} \mathbf{Y}_{n-1} \\ y_n \\ hy'_n \end{pmatrix}, \tag{2.18a}$$

where $M(x)$ is the $(s+2) \times (s+2)$ matrix defined by

$$M(z) = \begin{pmatrix} xA & \mathbf{e} & \mathbf{c} \\ x^2\mathbf{b}^\top A & 1 + x\mathbf{b}^\top\mathbf{e} & 1 + x\mathbf{b}^\top\mathbf{c} \\ x^2\mathbf{d}^\top A & x\mathbf{d}^\top\mathbf{e} & 1 + x\mathbf{d}^\top\mathbf{c} \end{pmatrix}. \tag{2.18b}$$

The matrix $M(x)$ will be called the *amplification matrix*, and its spectral radius $\rho(M(x))$ the *stability function*. For zero-stability, we have to demand that no eigenvalue of the matrix $M(0)$ has modulus greater than one, and that every eigenvalue of modulus one has multiplicity not greater than two. Thus the following theorem holds.

THEOREM 2.3. *EPTRKN methods based on any collocation vector* $\mathbf{c}$ *with distinct abscissas, are always zero-stable.*

The stability regions of EPTRKN methods are given by

$$(-\beta_{\text{stab}}, 0) := \{x : \rho(M(x)) \leq 1\}.$$

We shall call β_{stab} the *stability boundary* of EPTRKN methods (cf., e.g., [17]). The construction of EPTRKN methods possessing large stability regions will be considered in a forthcoming paper. The stability boundaries of a number of specified EPTRKN methods used in the numerical experiments are reported in Section 3.

2.3. On the Choice of Method Parameters

In the application of EPTRKN methods, for any chosen collocation vector $\mathbf{c}$ with distinct abscissas, the method parameter matrix A can be easily determined using the explicit expressions (2.11), the parameter vectors $\mathbf{b}$ and $\mathbf{d}$ are taken from the generating collocation IRKN methods. Since the class of collocation IRKN methods contains methods of arbitrarily high order (cf. [17,20]), we can obtain stable EPTRKN methods of arbitrarily high order. The freedom in the choice of the collocation points c_i of the collocation vector $\mathbf{c}$ can be used for obtaining some useful method properties. It seems natural to choose the abscissas of the vector $\mathbf{c}$ such

that the resulting EPTRKN method has the highest possible order $s+2$ like method (2.15d) (cf. Theorem 2.2) or good stability behaviour, etc. An other option minimizes the magnitudes of some components of the principal error vector in (2.5) defined by (2.6a) for the stage vector $\mathbf{Y}_n$. In this paper, we confine our considerations to the latter option. Although we may use (2.6a) for minimizing these magnitudes, it is more convenient to use the usual expression for the remainder term in Lagrange interpolation formulas. Using the r-point Lagrange interpolation formula for sufficiently differentiable functions $y(t)$ (see, e.g., [21, formulas 25.2.1–25.2.3])

$$\begin{aligned} y''(t_n+\tau h) &= \sum_{i=1}^{r} l_i(\tau) y''(t_n+a_i h) + C^{(r)}(\tau)\left(h\frac{d}{dt}\right)^r y(t_\tau^*), \\ l_i(\tau) &= \prod_{j=1, j\neq i}^{r} \frac{(\tau-a_j)}{(a_i-a_j)}, \qquad C^{(r)}(\tau) = \frac{1}{r!}\prod_{j=1}^{r}(\tau-a_j), \end{aligned} \tag{2.19}$$

where, t_τ^* is a suitably chosen point in the interval $[t_n, t_n+\tau h]$, we have the following alternative form of (2.5) (with $q=s+1$):

$$\begin{aligned} y(t_n+c_k h) = y(t_n) + h y_n' c_k + h^2 \sum_{i=1}^{s} l_i(1+c_k) y''(t_n+(c_i-1)h) \\ + C^{(s+2)}(1+c_k)\left(h\frac{d}{dt}\right)^{s+2} y(t_k^*), \qquad k=1,\ldots,s, \end{aligned} \tag{2.20}$$

where, t_k^* is also a suitably chosen point in the interval $[t_{n-1}, t_{n+1}]$. The principal error vector in (2.5) is given by $\mathbf{C}^{(s+2)} = C^{(s+2)}(\mathbf{e}+\mathbf{c})$, so that we are led to minimize the magnitude of the values

$$C^{(s+2)}(1+c_k) = \frac{1}{(s+1)!}\prod_{i=1}^{s}(1+c_k-c_i), \qquad k=1,2,\ldots.$$

From (2.21) we see that $C^{(s+2)}(1+c_k)$ vanishes if the set of components of the vector $\mathbf{c}$ contains the component c_k+1 (cf., e.g., [8,9]). Thus, we have the following theorem.

THEOREM 2.4. *Let $\mathbf{c}_v$ be an v-dimensional ($v<s$) subvector of the vector $\mathbf{c}$, and $\mathbf{e}_v$ be the v-dimensional vector with unit entries. If the set of components of the vector $\mathbf{c}$ contains the set of components of the vector $\mathbf{c}_v+\mathbf{e}_v$, then the stage subvector $\mathbf{Y}_n^v \approx y(t_n\mathbf{e}_v+\mathbf{c}_v h)$ has vanishing principal error subvector $C^{(s+2)}(\mathbf{e}_v+\mathbf{c}_v)$.*

In the numerical experiments presented in this paper, we confine our consideration to the EPTRKN methods based on the collocation vector $\mathbf{c}$ which satisfy the conditions of Theorem 2.4. We do not claim that this choice of method parameters is the best possible. A further study of this topic will be the subject of future research.

3. NUMERICAL EXPERIMENTS

In this section, we shall report the numerical results obtained by a number of new EPTRKN methods investigated in this paper and by a number of parallel and sequential methods from the literature. For the numerical experiments, we consider the following specified EPTRKN methods:

- EPTRKN3: third-order EPTRKN method based on $\mathbf{c} = (0, 1/2, 3/2)^\top$,
- EPTRKN4: fourth-order EPTRKN method based on $\mathbf{c} = (0, 1/2, 1, 3/2)^\top$,
- EPTRKN5: fifth-order EPTRKN method based on $\mathbf{c} = (0, 1/3, 2/3, 4/3, 5/3)^\top$,
- EPTRKN6: sixth-order EPTRKN method based on $\mathbf{c} = (0, 1/3, 2/3, 1, 4/3, 5/3)^\top$,
- EPTRKN7: seventh-order EPTRKN method based on $\mathbf{c} = (0, 1/4, 1/2, 1, 3/4, 5/4, 7/4)^\top$,
- EPTRKN8: eighth-order EPTRKN method based on $\mathbf{c} = (0, 1/4, 1/2, 3/4, 1, 5/4, 3/2, 7/4)^\top$,

- EPTRKN9: ninth-order EPTRKN method based on $\mathbf{c} = (-2/3, -1/3, 0, 1/3, 2/3, 1, 4/3, 5/3, 2)^T$,
- EPTRKN10: tenth-order EPTRKN method based on $\mathbf{c} = (-2/3, -1/2, -1/3, 1/3, 1/2, 2/3, 4/3, 3/2, 5/3)^T$.

Table 2. Stability boundaries β_{stab} for various p-order EPTRKN methods.

p-Order EPTRKN Method	$p=3$	$p=4$	$p=5$	$p=6$	$p=7$	$p=8$	$p=9$	$p=10$
β_{stab}	0.765	0.707	0.656	0.628	0.607	0.595	0.588	0.591

We numerically computed the stability boundaries β_{stab} as defined in Section 2.2 of these EPTRKN methods and reported them in Table 2. In terms of comparing stability of methods, it is the scaled stability region and not the stability region that is significant (cf., e.g., [22, p. 198]). The stability region of an EPTRKN method is at the same time the scaled stability region. Since the solution of problems of form (1.1) usually requires a stringent accuracy, we see from Table 2 that the stability regions of the above specified EPTRKN methods are good enough for nonstiff problems of this form.

We once again recall that the above-selected EPTRKN methods are not optimal. They are chosen by verifying only the conditions of Theorem 2.4. Moreover, in the numerical experiments, at first step, the starting values of $\mathbf{Y}_0$ and y_1 of an EPTRKN method will be generated by the associated PIRKN method using the direct collocation RKN corrector based on the same collocation vector $\mathbf{c}$ as the underlying EPTRKN method. All the computations were carried out on a 14-digit precision computer. In the tables of numerical results, negative NCDs are denoted by $(*)$. Furthermore, because of round-off errors, we cannot expect 14 digits accuracy. As a consequence, the tables of numerical results do contain empty spots whenever the corresponding result was in the neighbourhood of the accuracy-limits of the machine, and therefore, considered as unreliable.

For an easy comparison of the various methods, the (fixed) stepsize is chosen such that the total number of sequential **f**-evaluations of each method (approximately) equals a prescribed number N_{seq}. To be more precise, let s^* denote the number of sequential **f**-evaluations per step, N_{stp} denote the total number of integration steps for the integration interval $[t_0, T]$, then $N_{\text{seq}} = N_{\text{stp}} \cdot s^*$ which leads us to define

$$N_{\text{stp}} = \left[\frac{N_{\text{seq}}}{s^*} + \frac{1}{2}\right], \qquad h = \frac{T - t_0}{N_{\text{stp}}},$$

where $[\cdot]$ denotes the integer part function (the effect of the integer part operation causes that the actual number of sequential **f**-evaluations may be slightly different from the prescribed number N_{seq}).

Ignoring load balancing factors and communication times between processors in parallel methods, the comparison of various methods in this section is based on the numbers of sequential **f**-evaluations and on the obtained NCDs. The numerical experiments with small widely-used test problems taken from the literature below give a potential superiority of the new investigated methods over other extant methods. This superiority will be significant in a parallel environment if the test problems are large enough and/or the **f**-evaluations are computationally intensive enough. In this case, the communication overhead would not dominate the total computing time (cf., e.g., [22, p. 218]). An actual implementation with a stepsize strategy for large-expensive problems on a parallel machine is a subject of further studies.

3.1. Comparison with Parallel Methods

In this section, we report numerical results obtained by the best parallel explicit RKN methods available in the literature, that is the direct and indirect PIRKN methods proposed in [7,11] and

the EPTRKN methods specified above. Numerical experiments have shown that for the fixed number of iterations strategy, the efficiency of direct and indirect PIRKN methods are identical. The test set of three problems used in the numerical experiments are taken from the literature.

3.1.1. Nonlinear Fehlberg problem

For the first numerical test, we apply the various p^{th}-order PIRKN and EPTRKN methods to the well-known *nonlinear Fehlberg problem* (cf., e.g., [1,2,23,24])

$$\frac{d^2\mathbf{y}(t)}{dt^2} = \begin{pmatrix} -4t^2 & -\dfrac{2}{\sqrt{y_1^2(t)+y_2^2(t)}} \\ \dfrac{2}{\sqrt{y_1^2(t)+y_2^2(t)}} & -4t^2 \end{pmatrix} \mathbf{y}(t), \tag{3.1}$$

$$\mathbf{y}(0) = (0,1)^\top, \qquad \mathbf{y}'(0) = \left(-2\sqrt{\frac{\pi}{2}}, 0\right)^\top, \qquad \sqrt{\frac{\pi}{2}} \le t \le 10,$$

with highly oscillating exact solution given by $\mathbf{y}(t) = (\cos(t^2), \sin(t^2))^\top$. The results are reported in Table 3. These numerical results show that the EPTRKN methods are superior to the PIRKN methods of the same order by a speed-up factor about 4.

Table 3. NCD-values for several values of N_{seq} for problem (3.1) obtained by various p^{th}-order parallel methods.

Methods	p	$N_{\text{seq}} = 200$	$N_{\text{seq}} = 400$	$N_{\text{seq}} = 800$	$N_{\text{seq}} = 1600$	$N_{\text{seq}} = 3200$
PIRKN	3	0.1	0.6	1.4	2.4	3.3
EPTRKN3	3	1.3	2.1	3.0	3.9	4.8
PIRKN	4	0.1	1.1	2.4	3.5	4.7
EPTRKN4	4	2.3	3.6	4.9	6.1	7.4
PIRKN	5	0.1	1.7	3.3	4.9	6.5
EPTRKN5	5	3.1	4.7	6.3	7.8	9.3
PIRKN	6	0.2	1.9	3.8	5.7	7.4
EPTRKN6	6	4.6	6.3	8.2	10.0	11.8
PIRKN	7	0.2	2.6	4.9	7.2	9.4
EPTRKN7	7	5.6	8.3	10.4	12.4	
PIRKN	8	0.3	2.7	5.1	7.6	9.9
EPTRKN8	8	6.3	9.5	11.8		
PIRKN	9	0.4	3.4	6.4	9.4	11.7
EPTRKN9	9	7.0	10.4			
PIRKN	10	0.4	3.5	6.3	9.4	12.2
EPTRKN10	10	6.7	10.3			

3.1.2. Newton's equation of motion problem

The second numerical example is the two-body gravitational problem for Newton's equation of motion (see [25, p. 245]):

$$\frac{d^2y_1(t)}{dt^2} = -\frac{y_1(t)}{\left(\sqrt{y_1^2(t)+y_2^2(t)}\right)^3}, \qquad \frac{d^2y_2(t)}{dt^2} = -\frac{y_2(t)}{\left(\sqrt{y_1^2(t)+y_2^2(t)}\right)^3}, \qquad 0 \le t \le 20,$$

$$y_1(0) = 1-\varepsilon, \qquad y_2(0) = 0, \qquad y_1'(0) = 0, \qquad y_2'(0) = \sqrt{\frac{1+\varepsilon}{1-\varepsilon}}. \tag{3.2}$$

Table 4. NCD-values for several values of N_{seq} for problem (3.1) obtained by various p^{th}-order parallel methods.

Methods	p	$N_{\text{seq}} = 1600$	$N_{\text{seq}} = 3200$	$N_{\text{seq}} = 6400$	$N_{\text{seq}} = 12800$	$N_{\text{seq}} = 25600$
PIRKN	3	0.1	0.1	0.6	1.4	2.3
EPTRKN3	3	0.8	1.2	2.0	2.9	3.8
PIRKN	4	(*)	0.9	2.3	3.7	5.0
EPTRKN4	4	1.1	2.3	3.5	4.7	6.0
PIRKN	5	0.5	1.0	2.3	3.8	5.3
EPTRKN5	5	1.8	4.1	5.6	6.8	8.2
PIRKN	6	0.9	3.3	4.3	6.5	9.5
EPTRKN6	6	2.3	4.2	6.0	7.8	9.6
PIRKN	7	0.6	1.6	3.8	5.9	8.0
EPTRKN7	7	3.5	6.6	9.2	11.2	
PIRKN	8	2.0	3.1	5.5	8.1	10.7
EPTRKN8	8	3.7	6.2	8.6	10.9	
PIRKN	9	1.1	2.8	5.3	7.7	10.4
EPTRKN9	9	3.7	7.0	9.8	12.0	
PIRKN	10	1.0	3.5	6.1	9.6	12.5
EPTRKN10	10	3.5	9.0	11.7		

Table 5. NCD-values for several values of N_{seq} for problem (3.2) obtained by various p^{th}-order parallel PC methods.

Methods	p	$N_{\text{seq}} = 100$	$N_{\text{seq}} = 200$	$N_{\text{seq}} = 400$	$N_{\text{seq}} = 800$	$N_{\text{seq}} = 1600$
PIRKN	3	(*)	(*)	0.7	1.6	2.5
EPTRKN3	3	0.2	1.2	2.1	3.0	3.9
PIRKN	4	(*)	0.2	1.4	2.6	3.8
EPTRKN4	4	1.5	2.7	4.0	5.2	6.4
PIRKN	5	(*)	1.0	2.7	4.3	5.9
EPTRKN5	5	2.7	4.2	5.7	7.2	8.8
PIRKN	6	0.4	1.6	3.2	5.0	6.8
EPTRKN6	6	3.9	5.7	7.6	9.4	11.2
PIRKN	7	0.2	3.7	6.0	7.9	9.7
EPTRKN7	7	7.4	9.3	11.3		
PIRKN	8	0.5	2.8	5.0	7.3	9.7
EPTRKN8	8	6.9	9.1	11.5		
PIRKN	9	0.8	4.0	6.6	9.5	11.6
EPTRKN9	9	8.9	11.5			
PIRKN	10	0.7	4.2	6.7	9.6	12.4
EPTRKN10	10	8.5	11.4			

This problem can also be found in [24] or from the test set of problems in [26]. The solution components are $y_1(t) = \cos(u(t)) - \varepsilon$, $y_2(t) = \sqrt{(1+\varepsilon)(1-\varepsilon)}\,\sin(u(t))$, where $u(t)$ is the solution of Keppler's equation $t = u(t) - \varepsilon\,\sin(u(t))$ and ε denotes the eccentricity of the orbit. In this example, we set $\varepsilon = 0.9$. The results for this problem are given in Table 4 and show nearly the same superiority of the EPTRKN methods over the PIRKN methods as in the first example for the Fehlberg problem.

3.1.3. Scalar problem

The third example is the *scalar problem* taken from [23]

$$\frac{d^2y(t)}{dt^2} = -25y(t) + 100\cos(5t), \qquad y(0) = 1, \quad y'(0) = 5, \quad 0 \le t \le 10. \tag{3.3}$$

Table 6. NCD-values for several values of N_{seq} for problem (3.1) obtained by various p^{th}-order methods.

Methods	p	$N_{\text{seq}} = 400$	$N_{\text{seq}} = 800$	$N_{\text{seq}} = 1600$	$N_{\text{seq}} = 3200$	$N_{\text{seq}} = 6400$
N4	4	0.6	1.8	3.0	4.2	5.4
EPTRKN4	4	3.6	4.9	6.1	7.4	8.6
H8	8	0.3	2.6	5.2	7.6	10.0
BG8	8	0.9	3.1	5.6	8.0	10.4
EPTRKN8	8	9.5	11.8			
H10	10	(*)	2.2	5.4	8.5	11.5
EPTRKN10	10	10.3				

Table 7. NCD-values for several values of N_{seq} for problem (3.1) obtained by various p^{th}-order methods.

Methods	p	$N_{\text{seq}} = 3200$	$N_{\text{seq}} = 6400$	$N_{\text{seq}} = 12800$	$N_{\text{seq}} = 25600$	$N_{\text{seq}} = 51200$
N4	4	0.1	1.1	2.4	3.8	5.1
EPTRKN4	4	2.3	3.5	4.7	6.0	7.2
H8	8	0.1	2.0	4.6	7.4	10.6
BG8	8	0.6	3.1	6.5	8.2	10.5
EPTRKN8	8	6.2	8.6	10.9		
H10	10	0.3	1.8	4.9	8.4	11.7
EPTRKN10	10	9.0	11.7			

Table 8. NCD-values for several values of N_{seq} for problem (3.2) obtained by various p^{th}-order methods.

Methods	p	$N_{\text{seq}} = 200$	$N_{\text{seq}} = 400$	$N_{\text{seq}} = 800$	$N_{\text{seq}} = 1600$	$N_{\text{seq}} = 3200$
N4	4	(*)	0.9	2.1	3.3	4.5
EPTRKN4	4	2.7	4.0	5.2	6.4	7.6
H8	8	0.5	2.6	5.0	7.4	9.8
BG8	8	2.6	3.2	5.5	7.8	10.2
EPTRKN8	8	9.1	11.5			
H10	10	0.6	2.7	5.7	8.7	11.7
EPTRKN10	10	11.4				

The exact solution $y(t) = \cos(5t) + \sin(5t) + 10t\sin(5t)$ possesses rapidly oscillating components which are appearing with small and large variable amplitudes. The results for this problem are given in Table 5 and give rise to roughly the same conclusions as formulated in the two previous examples.

3.2. Comparison with Sequential Methods

In Section 3.1, a number of EPTRKN methods was compared with PIRKN methods (the most efficient parallel explicit RKN methods). In this section, some EPTRKN methods from them will be compared with some sequential explicit RKN methods currently available, that is

- N4: fourth-order first explicit RKN method constructed by Nyström in [27],
- H8: eighth-order explicit RKN method of Hairer from [4],
- BG8: eighth-order explicit RKN method of Beentjes and Gerritsen from [28],
- H10: tenth-order explicit RKN method constructed by Hairer in [5].

We shall report numerical results obtained by applying these four explicit RKN methods and the EPTRKN methods used in Section 3.1 of the same order to the three problems used also in Section 3.1. The numerical results of sequential explicit RKN methods listed in Tables 6–8 are

reproduced from the results given in [11]. The numerical results in these tables, show that the EPTRKN methods are by far superior to the sequential explicit RKN methods of the same order.

4. CONCLUDING REMARKS

This paper proposed a new general class of explicit pseudo two-step RKN methods requiring only *one* sequential f-evaluations per step for any order of accuracy. Implemented with fixed stepsize strategy, the specified explicit pseudo two-step RKN methods of order from 3 to 10 derived from this general class of methods are shown to be by far superior to the most efficient sequential and parallel methods currently available in the literature. These conclusions encourage us to pursue the study of EPTRKN methods. In particular, we will concentrate on the optimal choice of the method parameters and variable stepsize strategy implementations on parallel computers.

REFERENCES

1. E. Fehlberg, Klassische Runge-Kutta-Nyström-Formeln mit Schrittweitenkontrolle für Differentialgleichungen $x'' = f(t, x)$, *Computing* **10**, 305–315, (1972).
2. E. Fehlberg, Eine Runge-Kutta-Nyström-Formel 9-ter Ordnung mit Schrittweitenkontrolle für Differentialgleichungen $x'' = f(t, x)$, *Z. Angew. Math. Mech.* **61**, 477–485, (1981).
3. E. Fehlberg, S. Filippi and J. Gräf, Ein Runge-Kutta-Nyström-Formelpaar der Ordnung 10(11) für Differentialgleichungen $y'' = f(t, y)$, *Z. Angew. Math. Mech.* **66**, 265–270, (1986).
4. E. Hairer, Methodes de Nyström pour l'équations differentielle $y''(t) = f(t, y)$, *Numer. Math.* **27**, 283–300, (1977).
5. E. Hairer, A one-step method of order 10 for $y''(t) = f(t, y)$, *IMA J. Numer. Anal.* **2**, 83–94, (1982).
6. N.H. Cong, An improvement for parallel-iterated Runge-Kutta-Nyström methods, *Acta Math. Viet.* **18**, 295–308, (1993).
7. N.H. Cong, Note on the performance of direct and indirect Runge-Kutta-Nyström methods, *J. Comput. Appl. Math.* **45**, 347–355, (1993).
8. N.H. Cong, Explicit symmetric Runge-Kutta-Nyström methods for parallel computers, *Computers Math. Applic.* **31** (2), 111–122, (1996).
9. N.H. Cong, Explicit parallel two-step Runge-Kutta-Nyström methods, *Computers Math. Applic.* **32** (3), 119–130, (1996).
10. N.H. Cong, RKN-type parallel block PC methods with Lagrange-type predictors, *Computers Math. Applic.* **35** (9), 45–57, (1998).
11. B.P. Sommeijer, Explicit, high-order Runge-Kutta-Nyström methods for parallel computers, *Appl. Numer. Math.* **13**, 221–240, (1993).
12. N.H. Cong and T. Mitsui, Collocation-based two-step Runge-Kutta methods, *Japan J. Indust. Appl. Math.* **13**, 171–183, (1996).
13. N.H. Cong and T. Mitsui, A class of explicit parallel two-step Runge-Kutta methods, *Japan J. Indust. Appl. Math.* **14**, 303–313, (1997).
14. N.H. Cong, Direct collocation-based two-step Runge-Kutta-Nyström methods, *SEA Bull. Math.* **19**, 49–58, (1995).
15. N.H. Cong, Explicit pseudo two-step Runge-Kutta methods for parallel computers, *Intern. J. Computer Math.* **75** (to appear).
16. N.H. Cong, K. Strehmel and R. Weiner, Runge-Kutta-Nyström-type parallel block predictor-corrector methods, *Advances in Computational Mathematics* **10**, 115–133, (1999).
17. P.J. van der Houwen, B.P. Sommeijer and N.H. Cong, Stability of collocation-based Runge-Kutta-Nyström methods, *BIT* **31**, 469–481, (1991).
18. E. Hairer, S.P. Nørsett and G. Wanner, *Solving Ordinary Differential Equations, I. Nonstiff Problems*, Second Revised Edition, Springer-Verlag, Berlin, (1993).
19. N.H. Cong, Explicit pseudo two-step RKN methods with stepsize control, (submitted).
20. E. Hairer, Unconditionally stable methods for second order differential equations, *Numer. Math.* **32**, 373–379, (1979).
21. M. Abramowitz and I.A. Stegun, *Handbook of Mathematical Functions*, National Bureau of Standards Applied Mathematics Series 55, Dover, New York, (1970).
22. K. Burrage, *Parallel and Sequential Methods for Ordinary Differential Equations*, Clarendon Press, Oxford, (1995).
23. S. Filippi and J. Gräf, Ein Runge-Kutta-Nyström-Formelpaar der Ordnung 11(12) für Differentialgleichungen der Form $y'' = f(t, y)$, *Computing* **34**, 271–282, (1985).
24. S. Filippi and J. Gräf, New Runge-Kutta-Nyström formula-pairs of order 8(7), 9(8), 10(9) and 11(10) for differential equations of the form $y'' = f(t, y)$, *J. Comput. Appl. Math.* **14**, 361–370, (1986).
25. L.F. Shampine and M.K. Gordon, *Computer Solution of Ordinary Differential Equations, The Initial Value Problems*, W.H. Freeman and Company, San Francisco, CA, (1975).

26. T.E. Hull, W.H. Enright, B.M. Fellen and A.E. Sedgwick, Comparing numerical methods for ordinary differential equations, *SIAM J. Numer. Anal.* **9**, 603–637, (1972).
27. E.J. Nyström, Über die numerische Integration von Differentialgleichungen, *Acta Soc. Sci. Fenn.* **50** (13), 1–54, (1925).
28. P.A. Beentjes and W.J. Gerritsen, Higher order Runge-Kutta methods for the numerical solution of second order differential equations without first derivative, Report NW 34/76, Centre for Mathematics and Computer Science, Amsterdam, (1976).