Mathematical Modelling and Analysis
Volume 18 Number 2, April 2013, 289–307
http://dx.doi.org/10.3846/13926292.2013.785039

Publisher: Taylor&Francis and VGTU
http://www.tandfonline.com/TMMA
Print ISSN: 1392-6292
Online ISSN: 1648-3510

Implementation of explicit Nordsieck methods with inherent quadratic stability

Michał Braś[a,b,*], **Angelamaria Cardone**[c] **and Raffaele D'Ambrosio**[c]

[a] *AGH University of Science and Technology, Faculty of Applied Mathematics*
al. Mickiewicza 30, 30-059 Krakow, Poland
[b] *Jagiellonian University, Faculty of Mathematics and Computer Science*
ul. prof. S. Łojasiewicza 6, 30-348 Krakow, Poland
[c] *Università degli Studi di Salerno*
via Ponte Don Melillo, I-84084 Fisciano (SA), Italy
E-mail(*corresp.*): bras@agh.edu.pl
E-mail: ancardone@unisa.it
E-mail: rdambrosio@unisa.it

Received September 14, 2012; revised March 8, 2013; published online April 1, 2013

Abstract. The present paper deals with the implementation in a variable-step algorithm of general linear methods in Nordsieck form with inherent quadratic stability and large stability regions constructed recently by Braś and Cardone. Various implementation issues such as rescale strategy, local error estimation, step-changing strategy and starting procedure are discussed. Some numerical experiments are reported, which show the performances of the methods and make comparisons with other existing methods.

Keywords: general linear methods, quadratic stability, implementation.

AMS Subject Classification: 65L05, 65L20.

1 Introduction

We consider initial value problem for the system of ordinary differential equations (ODEs) in the autonomous form:

$$\begin{cases} y' = f(y), & t \in [t_0, T], \\ y(t_0) = y_0, \end{cases} \tag{1.1}$$

* The work of this author was supported by the National Science Center under grant DEC-2011/01/N/ST1/02672 and by the Polish Ministry of Science and Higher Education.

where $f\colon \mathbb{R}^d \to \mathbb{R}^d$ is a sufficiently smooth function. Consider the uniform grid $t_n = t_0 + nh,\ n = 0, 1, \ldots, N,\ h = (T - t_0)/N$. The general linear method (GLM) in Nordsieck form with coefficient matrices $\mathbf{A} \in \mathbb{R}^{s\times s}$, $\mathbf{U} \in \mathbb{R}^{s\times r}$, $\mathbf{B} \in \mathbb{R}^{r\times s}$, $\mathbf{V} \in \mathbb{R}^{s\times s}$ and abscissa vector $\mathbf{c} \in \mathbb{R}^s$ is defined by

$$\begin{cases} Y^{[n]} = h\,(\mathbf{A}\otimes I)\, f(Y^{[n]}) + (\mathbf{U}\otimes I)\, z^{[n-1]}, \\ z^{[n]} = h\,(\mathbf{B}\otimes I)\, f(Y^{[n]}) + (\mathbf{V}\otimes I)\, z^{[n-1]}, \end{cases} \tag{1.2}$$

$n = 1, 2 \ldots, N$, where I is the identity matrix of dimension d,

$$Y^{[n]} = \begin{bmatrix} Y_1^{[n]} \\ \vdots \\ Y_s^{[n]} \end{bmatrix}, \quad f(Y^{[n]}) = \begin{bmatrix} f(Y_1^{[n]}) \\ \vdots \\ f(Y_s^{[n]}) \end{bmatrix}, \quad z^{[n]} = \begin{bmatrix} z_1^{[n]} \\ \vdots \\ z_r^{[n]} \end{bmatrix},$$

s is the number of internal stages and r is the number of input and output approximations. $Y_i^{[n]}$ is an approximation of stage order q to $y(t_n + c_i h)$, i.e.,

$$Y_i^{[n]} = y(t_{n-1} + c_i h) + O(h^{q+1}), \quad i = 1, 2, \ldots, s,$$

$z_i^{[n]}$ is an approximation of order p to the component $h^{i-1}y^{(i-1)}(t_n)$ of the Nordsieck vector, i.e., if

$$z_i^{[n-1]} = h^{i-1}y^{(i-1)}(t_{n-1}) + O(h^{p+1}),$$

then

$$z_i^{[n]} = h^{i-1}y^{(i-1)}(t_n) + O(h^{p+1}), \quad i = 1, 2, \ldots, r.$$

From the theory of order and stage order conditions for GLMs, the following theorem follows. We omit the proof, and the reader can refer, for example to [19].

Theorem 1. *The GLM* (1.2) *has order* $p = q$ *if and only if*

$$e^{\mathbf{c}z} = z\mathbf{A}e^{\mathbf{c}z} + \mathbf{U}Z + O(z^{p+1}), \tag{1.3}$$

$$e^{z}Z = z\mathbf{B}e^{\mathbf{c}z} + \mathbf{V}Z + O(z^{p+1}), \tag{1.4}$$

where $e^{\mathbf{c}z} = [\ e^{c_1 z} \quad e^{c_2 z} \quad \cdots \quad e^{c_s z}\ ]^T$ *and* $Z = [\ 1 \quad z \quad \cdots \quad z^{r-1}\ ]^T$.

By suitable series expansion of order conditions (1.3) and (1.4), algebraic conditions on the coefficient matrices can been derived, as done in [4,9].

Here we consider GLMs in Nordsieck form with $s = r-1$, and the coefficient matrices $\mathbf{A}$ and $\mathbf{V}$ are assumed to be of the form

$$\mathbf{A} = \begin{bmatrix} 0 & & & & \\ a_{21} & 0 & & & \\ a_{31} & a_{32} & \ddots & & \\ \vdots & \vdots & \ddots & \ddots & \\ a_{s,1} & a_{s,2} & \cdots & a_{s,s-1} & 0 \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} 1 & v_{12} & v_{13} & \cdots & v_{1r} \\ 0 & 0 & v_{23} & \cdots & v_{2r} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & & \ddots & v_{r-1,r} \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix},$$

so the resulting method is explicit and zero-stable. Nordsieck representation of DIMSIMs was discussed in [6] and that of two-step Runge-Kutta methods was derived in [1].

In these paper we deal with the implementation issues related to the methods with $p = q = s = r - 1$, $p = 1, 2, \ldots, 5$ constructed in [4], which satisfy the property of inherent quadratic stability (IQS), compare [3,4,9,10,11]. This property guarantees that the stability polynomial of the GLM takes the form

$$p(\omega, z) = \omega^{r-2} \left(\omega^2 - p_1(z)\omega + p_0(z)\right),$$

thus its stability properties depend on the quadratic polynomial

$$\widetilde{p}(\omega, z) = \omega^2 - p_1(z)\omega + p_0(z).$$

Such a property is useful for the practical derivation of highly stable methods (e.g. A- and L-stable) in the implicit case (compare [3, 11, 19]), and methods with large stability regions in the explicit one case (see [4, 9, 10]).

The paper is organized as follows. Section 2 is concerned with the variable stepsize formulation of Nordsieck methods and a reliable error estimation, Section 3 deals with some issues necessary to implement such methods in a variable stepsize environment, such as a technique to update the vector of external approximations, a step control strategy and starting procedures. Section 4 provides the numerical evidence originated by implementing Nordsieck GLMs with fixed and variable stepsize, while Section 5 contains some conclusions and further developments of this research.

2 Variable step-size formulation of the methods

For the variable step formulation of the methods, it is useful to set

$$z^{[n]} = \begin{bmatrix} y_n \\ \overline{z}^{[n]} \end{bmatrix},$$

where y_n is an approximation to $y(t_n)$ and $\overline{z}^{[n]}$ is an approximation to $z(t_n, h_n)$, $h_n = t_n - t_{n-1}$, with

$$z(t, h) := \begin{bmatrix} hy'(t) \\ h^2 y''(t) \\ \vdots \\ h^p y^{(p)}(t) \end{bmatrix}.$$

Then GLM (1.2), on the nonuniform grid $t_0 < t_1 < \cdots < t_N, \quad t_N \geq T$, can be formulated as (compare [19])

$$\begin{cases} Y^{[n]} &= (\mathbf{e} \otimes I) y_{n-1} + h_n (A \otimes I) F(Y^{[n]}) + (U \otimes I) z^{[n-1]}, \\ y_n &= y_{n-1} + h_n (b^T \otimes I) F(Y^{[n]}) + (v^T \otimes I) z^{[n-1]}, \\ \overline{z}^{[n]} &= h_n (B \otimes I) F(Y^{[n]}) + (V \otimes I) z^{[n-1]}, \end{cases} \tag{2.1}$$

where

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{c|cc} A & \mathbf{e} & U \\ \hline b^T & 1 & v^T \\ B & 0 & V \end{array}\right],$$

$\mathbf{e} = [1, 1, \ldots, 1]^T \in \mathbb{R}^s$, $b \in \mathbb{R}^s$, $v \in \mathbb{R}^{r-1}$, $A \in \mathbb{R}^{s\times s}$, $U \in \mathbb{R}^{s\times(r-1)}$, $B \in \mathbb{R}^{(r-1)\times s}$, $V \in \mathbb{R}^{(r-1)\times(r-1)}$, I stands for the identity matrix of dimension d.

2.1 Error propagation

The following result analyzes the error propagation from one step to another (compare [19], Sec. 8.3, [5,8]). We report the proof, which follows the lines of Theorem 2.1 of [5], for sake of completeness.

Theorem 2. *Assume that the input quantities to the current step from t_{n-1} to $t_n = t_{n-1} + h_n$ satisfy*

$$\left\{\begin{array}{lcl} y_{n-1} & = & y(t_{n-1}), \\ z^{[n-1]} & = & z(t_{n-1}, h_n) - (\beta \otimes I) h_n^{p+1} y^{(p+1)}(t_{n-1}) + O(h_n^{p+2}), \end{array}\right. \tag{2.2}$$

where $y(t)$ is the solution to the differential system (1.1) *and require that*

$$\left\{\begin{array}{lcl} y_n & = & y(t_n) - E h_n^{p+1} y^{(p+1)}(t_n) + O(h_n^{p+2}) \\ \overline{z}^{[n]} & = & \overline{z}(t_n, h_n) - (\beta \otimes I) h_n^{p+1} y^{(p+1)}(t_n) + O(h_n^{p+2}) \end{array}\right. \tag{2.3}$$

with the same vector β. Here, $\overline{z}(t_n, h_n)$ is the Nordsieck vector corresponding to the solution $\overline{y}(t)$ of the initial value problem

$$\left\{\begin{array}{ll} \overline{y}'(t) & = f(\overline{y}(t)), \qquad t \in [t_n, t_{n+1}], \\ \overline{y}(t_n) & = y_n. \end{array}\right.$$

Then it follows that (2.3) *holds if*

$$\begin{aligned} &E = \frac{1}{(p+1)!} - \frac{b^T \mathbf{c}^p}{p!} + v^T \beta, \\ &\beta = (I - V)^{-1} \left(t_p - B \tfrac{\mathbf{c}^p}{p!} \right), \quad t_p = \left[\begin{array}{cccc} \frac{1}{p!} & \frac{1}{(p-1)!} & \cdots & 1 \end{array}\right]^T. \end{aligned} \tag{2.4}$$

Proof. From the hypothesis that $q = p$, it follows that

$$Y^{[n]} = y(t_{n-1} + \mathbf{c} h_n) - (\sigma \otimes I) h_n^{p+1} y^{(p+1)}(t_{n-1}) + O(h_n^{p+2}), \tag{2.5}$$

for a certain vector σ. Therefore,

$$\begin{aligned} h_n F(Y^{[n]}) = \; & h_n y'(t_{n-1} + \mathbf{c} h_n) - \\ & (\sigma \otimes I) h_n^{p+2} \frac{\partial f}{\partial y}(y(t_{n-1})) y^{(p+1)}(t_{n-1}) + O(h_n^{p+3}). \end{aligned} \tag{2.6}$$

Moreover

$$\begin{aligned} h\overline{y}'(t_n) &= h_n f(\overline{y}(t_n)) = hf(y_n) = hf(y(t_n) - Eh^{p+1}y^{(p+1)}(t_n)) + O(h_n^{p+2}) \\ &= hy'(t_n) + O(h_n^{p+2}), \\ h^k\overline{y}^{(k)}(t_n) &= h_n^k y^{(k)}(t_n) + O(h_n^{p+3}), \end{aligned}$$

therefore

$$\overline{z}(t_n, h_n) = z(t_n, h_n) + O(h_n^{p+2}). \tag{2.7}$$

Substituting (2.5), (2.6) and (2.7) into (2.3), and using the localizing assumptions given by (2.2), it follows

$$\begin{aligned} y(t_{n-1} + \mathbf{c}h_n) - (\xi \otimes I)h_n^{p+1}(t_{n-1}) &= (\mathbf{e} \otimes I)y_{n-1} + (A \otimes I)h_n y'(t_{n-1} + \mathbf{c}h_n) \\ &+ (U \otimes I)\left(z(t_{n-1}, h_n) - (\beta \otimes I)h_n^{p+1}y^{(p+1)}(t_{n-1})\right) + O(h_n^{p+2}), \end{aligned} \tag{2.8}$$

$$\begin{aligned} y(t_n) - Eh_n^{p+1}y^{(p+1)}(t_{n-1}) &= y(t_{n-1}) + (b^T \otimes I)h_n y'(t_{n-1} + \mathbf{c}h_n) \\ &+ (v^T \otimes I)\left(z(t_{n-1}, h_n) - (\beta \otimes I)h_n^{p+1}y^{(p+1)}(t_{n-1})\right) + O(h_n^{p+2}), \end{aligned} \tag{2.9}$$

$$\begin{aligned} z(t_n, h_n) - (\beta \otimes I)h_n^{p+1}y^{(p+1)}(t_{n-1}) &= (B \otimes I)h_n y'(t_{n-1} + \mathbf{c}h_n) \\ &+ (V \otimes I)\left(z(t_{n-1}, h_n) - (\beta \otimes I)h_n^{p+1}y^{(p+1)}(t_{n-1})\right) + O(h_n^{p+2}). \end{aligned} \tag{2.10}$$

By expanding in Taylor series $y(t_{n-1} + \mathbf{c}h_n)$ and $y'(t_{n-1} + \mathbf{c}h_n)$ around t_{n-1}, equating terms of (2.8) of order up to $p+1$ and considering that terms of order up to p cancel out by order conditions, from (2.8) and (2.9), we derive ξ as in (2.4). By expanding in Taylor series $y(t_n)$ around t_{n-1}, and considering that

$$y^{(p+1)}(t_n) = y^{(p+1)}(t_{n-1}) + h_n y^{(p+2)}(t_{n-1}) + O(h_n^2), \tag{2.11}$$

and successively equating terms of order up to $p+1$ in (2.9) (and considering again that terms of order up to p cancel out by order conditions), we get the coefficient E as in (2.4).
Last, by expanding in Taylor series $h_n^k y^{(k)}(t_n)$ around t_{n-1} into $z(t_n, h_n)$, it follows

$$z(t_n, h_n) = T_p z(t_{n-1}, h_n) + t_p h_n^{p+1}y^{(p+1)}(t_{n-1}) + O(h_n^2), \tag{2.12}$$

with t_p defined as in (2.4) and

$$T_p = \begin{bmatrix} 1 & 1 & \frac{1}{2!} & \cdots & \frac{1}{(p-1)!} \\ 0 & 1 & 1 & \cdots & \frac{1}{(p-2)!} \\ 0 & 0 & 1 & \cdots & \frac{1}{(p-3)!} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}.$$

Substituting (2.12) in (2.10), equating terms of order up to $p+1$, we obtain β as in (2.4). □

2.2 Local error estimation

It follows from Theorem 2 that the local error is

$$le(t_n) = Eh_n^{p+1} y^{(p+1)}(t_n) + O(h_n^{p+2}). \tag{2.13}$$

We look for estimation of the quantity $h_n^{p+1} y^{(p+1)}(t_n)$ in the form

$$h_n^{p+1} y^{(p+1)}(t_n) = (\varphi^T \otimes I) h_n F(Y^{[n]}) + (\psi^T \otimes I) z^{[n-1]} + O(h_n^{p+2}). \tag{2.14}$$

The following result holds (compare [5,19]). We report the proof for completeness.

Theorem 3. *Consider the GLM* (2.1) *of order* p *and stage order* $q = p$ *and assume that* f *is sufficiently smooth. The vectors* $\varphi \in \mathbb{R}^s$ *and* $\psi \in \mathbb{R}^{r-1}$ *in* (2.14) *satisfy the linear system*

$$\begin{cases} \varphi^T \dfrac{\mathbf{c}^{j-1}}{(j-1)!} + \psi_j = 0, \quad j = 1, 2, \ldots, r-1, \\ \varphi^T \dfrac{\mathbf{c}^p}{p!} - \psi^T \beta = 1. \end{cases} \tag{2.15}$$

Proof. Similarly as in [19], we will use expansion:

$$\begin{aligned} h_n y'(t_{n-1} + \mathbf{c} h_n) \quad &= \sum_{i=1}^{p} \frac{\mathbf{c}^{i-1} \otimes I}{(i-1)!} h_n^i y^{(i)}(t_{n-1}) \\ &+ \frac{\mathbf{c}^p \otimes I}{p!} h_n^{p+1} y^{(p+1)}(t_{n-1}) + O(h_n^{p+2}). \end{aligned} \tag{2.16}$$

By using (2.16) and (2.6) we get:

$$\begin{aligned} h_n F(Y^{[n]}) = \quad & \sum_{i=1}^{p} \frac{\mathbf{c}^{i-1} \otimes I}{(i-1)!} h_n^i y^{(i)}(t_{n-1}) + \frac{\mathbf{c}^p \otimes I}{p!} h_n^{p+1} y^{(p+1)}(t_{n-1}) \\ & + O(h_n^{p+2}). \end{aligned} \tag{2.17}$$

We substitute (2.11) on the left of (2.14), use localizing assumptions (2.2) and (2.17) on the right of (2.14), and we obtain

$$\begin{aligned} & h_n^{p+1} y^{(p+1)}(t_{n-1}) = \\ & \quad (\varphi^T \otimes I) \left(\sum_{i=1}^{p} \frac{\mathbf{c}^{i-1} \otimes I}{(i-1)!} h_n^i y^{(i)}(t_{n-1}) + \frac{\mathbf{c}^p \otimes I}{p!} h_n^{p+1} y^{(p+1)}(t_{n-1}) \right) \\ & \qquad + (\psi^T \otimes I) \Big(z(t_{n-1}, h_n) - (\beta \otimes I) h_n^{p+1} y^{(p+1)}(t_n) \Big) + O(h_n^{p+2}). \end{aligned}$$

Now, by equating the coefficients of terms of order up to $p+1$, the thesis follows. □

In our cases, $p = q = s = r - 1$, thus the linear system (2.15) consists of $p+1$ equations, so there are $p-1$ free parameters. For $p \geq 2$, we require that the contribution of both terms in the last equation is the same, i.e.,

$$\frac{\varphi^T \mathbf{c}^p}{p!} = \frac{1}{2}, \quad -\psi^T \beta = \frac{1}{2} \tag{2.18}$$

and the number of free parameters is reduced to $p-2$. From order $p=3$ we fix the remaining free parameters to zero, i.e.,

$$\psi_3 = \cdots = \psi_p = 0. \tag{2.19}$$

Different choices of $\psi_3, ..., \psi_p$ are possible as well, see for example [2].

Now we compute vectors φ, ψ for the methods with $p = q = s = r-1$, $p = 1, 2, \ldots, 5$, constructed in [4].

Example 1. $p = q = s = r-1 = 1$. Here, $\varphi, \psi \in \mathbb{R}$, and the system (2.15) takes the form

$$\begin{cases} \varphi + \psi = 0, \\ \varphi c_1 - \psi\beta = 1. \end{cases} \tag{2.20}$$

Taking into account that $c_1 = 1$, and by formula (2.6) $\beta = 0$, we obtain the unique solution $\varphi = 1$, $\psi = -1$.

Example 2. $p = q = s = r-1 = 2$. Equations (2.15) and (2.18) give

$$\begin{cases} \varphi^T \mathbf{e} + \psi_1 = 0, \\ \varphi^T \mathbf{c} + \psi_2 = 0, \\ \varphi^T \frac{\mathbf{c}^2}{2!} = \frac{1}{2}, \\ -\psi^T \beta = \frac{1}{2}, \end{cases} \tag{2.21}$$

where $\mathbf{e} = \mathbf{c}^0 = [1,1]^T$, $\mathbf{c} = [0,1]^T$, and $\beta = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \end{bmatrix}^T$. The unique solution is

$$\varphi = \begin{bmatrix} -1 & 1 \end{bmatrix}^T, \quad \psi = \begin{bmatrix} 0 & -1 \end{bmatrix}^T.$$

In analogous way, by (2.16) and (2.19), we derive the following vectors φ, ψ for higher order methods.

Example 3. $p = q = s = r-1 = 3$

$$\mathbf{c} = \begin{bmatrix} 0 & \frac{1}{2} & 1 \end{bmatrix}^T, \quad \beta = \begin{bmatrix} \frac{1}{6} & \frac{1}{3} & \frac{1}{2} \end{bmatrix}^T, \quad E = \frac{9503}{142992},$$
$$\varphi = \begin{bmatrix} 33 & -24 & 6 \end{bmatrix}^T, \quad \psi = \begin{bmatrix} -15 & 6 & 0 \end{bmatrix}^T.$$

Example 4. $p = q = s = r-1 = 4$

$$\mathbf{c} = \begin{bmatrix} 0 & \frac{1}{3} & \frac{2}{3} & 1 \end{bmatrix}^T, \quad \beta = \begin{bmatrix} \frac{1}{24} & \frac{1}{9} & \frac{29}{108} & \frac{1}{2} \end{bmatrix}^T, \quad E = \frac{26105531}{1632823920},$$
$$\varphi = \begin{bmatrix} -429 & 486 & -243 & 54 \end{bmatrix}^T, \qquad \psi = \begin{bmatrix} 132 & -54 & 0 & 0 \end{bmatrix}^T.$$

Example 5. $p = q = s = r-1 = 5$

$$\mathbf{c} = \begin{bmatrix} 0 & \frac{1}{4} & \frac{1}{2} & \frac{3}{4} & 1 \end{bmatrix}^T, \quad \beta = \begin{bmatrix} \frac{1}{120} & \frac{9}{320} & \frac{11}{128} & \frac{15}{64} & \frac{1}{2} \end{bmatrix}^T,$$
$$E = \frac{545203}{194357800}, \quad \varphi = \begin{bmatrix} \frac{22660}{3} & -10240 & 7680 & -\frac{10240}{3} & 640 \end{bmatrix}^T,$$
$$\psi = \begin{bmatrix} -2220 & 640 & 0 & 0 & 0 \end{bmatrix}^T.$$

2.3 Rescale strategy

After the step t_n is performed by (2.1), we have computed $\overline{z}^{[n]} \approx z(t_n, h_n)$. In order to perform next step, we need as a new input vector $z^{[n]} \approx z(t_n, h_{n+1})$, with $h_{n+1} = t_{n+1} - t_n$. The most natural way to get $z^{[n]}$ is to rescale the vector $z^{[n]}$, i.e.,

$$z^{[n]} = D(\delta)\overline{z}^{[n]}$$

with $D(\delta) = \mathrm{diag}(\delta, \delta^2, \ldots, \delta^s)$, and $\delta = h_{n+1}/h_n$. Other strategies are also available, see for example [19].

3 Implementation issues

This section is devoted to the description of the issues we have considered for the implementation of our methods in a variable stepsize environment.

3.1 Starting procedure

The implementation of Nordsieck methods needs the assessment of a suitable starting procedure for the derivation of the input vector associated to the initial stepsize. For simple problems, the derivatives contained in the input vector can be computed exactly from the given problem, since

$$y' = f, \quad y'' = f_y f, \quad y''' = f_{yy} f f + f_y f_y f, \quad \ldots$$

where the subscripts denote partial derivations with respect to y. Hence, the elements of the initial input vector $z^{[0]}$ can be exactly computed in terms of the elementary differentials of the problem.

In more general cases, the employ of an additional starting method for the computation of the initial input vector is required. Following [19, 21], we compute the vector $z^{[0]}$ by means of the formulas

$$\overline{Y}_i = y_0 + h \sum_{j=1}^{p} \overline{a}_{ij} f(\overline{Y}_j), \quad z_i^{[0]} = h \sum_{j=1}^{p} \overline{b}_{ij} f(\overline{Y}_j),$$

$i = 1, 2, \ldots, p$, where the matrices $\overline{A} = [\overline{a}_{ij}] \in \mathbb{R}^{p \times p}$ and $\overline{B} = [\overline{b}_{ij}] \in \mathbb{R}^{p \times p}$ are computed in such a way that

$$\overline{Y}_i = y(t_0 + \overline{c}_i h) + O(h^{p+1}), \quad z_i^{[0]} = h^i y^{(i)}(t_0) + O(h^{p+1}),$$

$i = 1, 2, \ldots, p$, being $\overline{\mathbf{c}} = [\overline{c}_1, \overline{c}_2, \ldots, \overline{c}_p]^T \in \mathbb{R}^p$ is a given abscissa vector for the starting method. The algebraic constraints on the coefficients of the starting procedure which guarantee the above accuracy requirements are (see [19, 21])

$$\sum_{j=1}^{p} \overline{a}_{ij} \frac{\overline{c}_j^{k-1}}{(k-1)!} = \frac{\overline{c}_i^k}{k!},$$

$$\sum_{j=1}^{p} \overline{b}_{ij} \frac{\overline{c}_j^{k-1}}{(k-1)!} = \delta_{ik},$$

$i,k = 1,2,\ldots,p$. In [19], Chapter 8, $\overline{A},\overline{B}$ are computed for equally spaced abscissa $\overline{\mathbf{c}}$, with $p = 1,\ldots,4$. For $p = 5$, and equally spaced abscissa $\overline{\mathbf{c}}$, the tableau of the coefficient matrices of the starting method assume the form

$$\left[\begin{array}{c|c} \mathbf{e} & \overline{A} \\ \hline \overline{\mathbf{c}} & \overline{B} \end{array}\right] = \left[\begin{array}{c|ccccc} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & \frac{251}{2880} & \frac{323}{1440} & -\frac{11}{120} & \frac{53}{1440} & -\frac{19}{2880} \\ 1 & \frac{29}{360} & \frac{31}{90} & \frac{1}{5} & \frac{1}{90} & -\frac{1}{360} \\ 1 & \frac{27}{320} & \frac{51}{160} & \frac{9}{40} & \frac{21}{160} & -\frac{3}{320} \\ 1 & \frac{7}{90} & \frac{16}{45} & \frac{2}{15} & \frac{16}{45} & \frac{7}{90} \\ \hline 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{1}{4} & -\frac{25}{3} & 16 & -12 & \frac{16}{3} & -1 \\ \frac{1}{2} & \frac{140}{3} & -\frac{416}{3} & 152 & -\frac{224}{3} & \frac{44}{3} \\ \frac{3}{4} & -160 & 576 & -768 & 448 & -96 \\ 1 & 256 & -1024 & 1536 & -1024 & 256 \end{array}\right].$$

This starting method, similarly as those presented in [19,21], is implicit. However, Huang [18] proposed different starting formulae based on singly diagonally implicit GLMs.

Regarding the choice of the initial stepsize of integration h_0, we assume that

$$h_0 = \min\left\{\frac{T-t_0}{N}, \frac{tol^{\frac{1}{p+1}}}{\|f(y_0)\|}\right\} \tag{3.1}$$

for given tolerance tol, being p the order of the method and $\|\cdot\|$ is the Euclidean norm. Here, N is an arbitrary number introduced in order to prevent the initial step form being too large; in our tests we set $N = 100$.

3.2 The necessity of a reliable error estimate

The derivation of a reliable error estimate is the first building block for the construction of an efficient strategy of control of the stepsize. In fact, after performing the current step with stepsize h_n, we have compared the requested accuracy tol with the achieved error estimate, by checking if the following inequality holds

$$\|est(n)\| \leq tol, \tag{3.2}$$

based on the error estimate above described, in order to compute the next value of the stepsize accordingly. If the control (3.2) is not satisfied, the stepsize h_{n+1} used in the next step is halved. Otherwise, the new stepsize h_{n+1} is suitably chosen, according to one of the following step changing strategy.

3.3 Step control strategies

A standard step control strategy (see [16])

$$h_{n+1} = h_n \cdot \min\left(2, \left(\frac{fac \cdot tol}{\|est(t_n)\|}\right)^{\frac{1}{p+1}}\right),$$

which only depends on the estimate computed in the previous step, can often determine useless stepsize rejections. Thus, Gustafsson, Lundh and Söderlind [15] introduced a different stepsize control, the so-called PI stepsize control, based on control theory arguments [5, 15, 17]. The PI control involves the estimation of the local errors related to the two most recent subintervals of the discretization, i.e.,

$$h_{n+1} = h_n \cdot \min\left(2, \left(\frac{tol}{\|est(t_n)\|}\right)^{\sigma_1}\left(\frac{tol}{\|est(t_{n-1})\|}\right)^{\sigma_2}\right),$$

where σ_1 and σ_2 are constant values which must be suitably chosen. As in [14, 17] we have experimentally found some values for σ_1 and σ_2 which make the PI stepsize control competitive with the standard strategy: such values are

$$\sigma_1 = 0.07/(p+1), \quad \sigma_2 = 1.2/(p+1), \tag{3.3}$$

where p the is order of the method.

4 Numerical experiments

4.1 Test problems

We consider the following test problems:

- linear test problem
$$y' = -\lambda y, \quad t \in [0, T], \tag{4.1}$$

- Prothero-Robinson type problem [20]
$$\begin{cases} y' = -16y + 15e^{-t}, & t \in [0, 100] \\ y(0) = 2 \end{cases} \tag{4.2}$$
with exact solution $y(t) = e^{-t} + e^{-16t}$.

- van der Pohl equation [17]
$$\begin{cases} y_1' = y_2, & y_1(0) = 2, \\ y_2' = \varepsilon(1 - y_1^2)y_2 - y_1, & y_2(0) = 0. \end{cases}, \tag{4.3}$$
$\varepsilon = 200$, $t \in [0, 20]$ (the reference solution is obtained with Matlab `ode15s` procedure).

4.2 Constant stepsize implementation

We apply methods of order $p = 1, 2, \ldots 5$ to linear test equation (4.1), with $\lambda = 40$, $T = 1$ and fixed stepsize $h = T/N$, where $N = 2^{i+1} \cdot 10$, $i = 1, 2, \ldots, 10$. We calculated the norm $||e_h(T)||$ of error at endpoint (here, error is the difference between exact solution and the one obtained by the method) and the effective order of convergence p_{eff} given by

$$p_{\text{eff}} = \frac{\log(||e_h(T)||/||e_{h/2}(T)||)}{\log(2)}. \tag{4.4}$$

The results are collected in Tab. 1, and they confirm the theoretical order of convergence.

Table 1. Numerical results of method of order $p = 1, 2, 3, 4, 5$ for problem (4.1), $T = 1$, with constant stepsize.

	$p=1$		$p=2$		$p=3$	
N	$\|\|e_h(T)\|\|$	p_{eff}	$\|\|e_h(T)\|\|$	p_{eff}	$\|\|e_h(T)\|\|$	p_{eff}
640	$3.62 \cdot 10^{-18}$		$1.66 \cdot 10^{-19}$		$3.34 \cdot 10^{-21}$	
1280	$2.55 \cdot 10^{-18}$	0.50	$4.01 \cdot 10^{-20}$	2.05	$4.07 \cdot 10^{-22}$	3.04
2560	$1.54 \cdot 10^{-18}$	0.73	$9.91 \cdot 10^{-21}$	2.02	$5.02 \cdot 10^{-23}$	3.02
5120	$8.49 \cdot 10^{-19}$	0.86	$2.46 \cdot 10^{-21}$	2.01	$6.23 \cdot 10^{-24}$	3.01
10240	$4.46 \cdot 10^{-19}$	0.93	$6.14 \cdot 10^{-22}$	2.00	$7.77 \cdot 10^{-25}$	3.00
20480	$2.29 \cdot 10^{-19}$	0.96	$1.53 \cdot 10^{-22}$	2.00	$9.69 \cdot 10^{-26}$	3.00
	$p=4$		$p=5$			
N	$\|\|e_h(T)\|\|$	p_{eff}	$\|\|e_h(T)\|\|$	p_{eff}		
640	$7.47 \cdot 10^{-22}$		$1.47 \cdot 10^{-23}$			
1280	$4.41 \cdot 10^{-23}$	4.08	$4.60 \cdot 10^{-25}$	5.00		
2560	$2.68 \cdot 10^{-24}$	4.04	$1.43 \cdot 10^{-26}$	5.01		
5120	$1.65 \cdot 10^{-25}$	4.02	$4.45 \cdot 10^{-28}$	5.01		
10240	$1.02 \cdot 10^{-26}$	4.01	$1.33 \cdot 10^{-29}$	5.06		
20480	$6.38 \cdot 10^{-27}$	4.01	$1.35 \cdot 10^{-31}$	6.62		

In Tab. 2 we compare the error of Nordsieck method with IQS of order $p = 4$ and method with IRKS of the same order, corresponding to $\eta = 3/5$ with error constant E = 1/300, whose coefficients are listed in [7]. We considered the test problem (4.1) on interval $[0, 10]$ for some chosen values of λ and also test problem (4.2). The results show that our methods converge for a larger value of stepsize with respect to IRKS methods. We notice that for the same order our methods are also cheaper, since IRKS method has $s = p+1$ internal stages (compare [19]), while our methods have $s = p$.

Table 2. Errors of Nordsieck method and IRKS method of order $p = 4$ for problem (4.1), with $T = 10$, and problem (4.2), with constant stepsize.

problem (4.1), $\lambda = 50$				problem (4.2)			
N	h	IQS $p = 4$	IRKS $p = 4$	N	h	IQS $p = 4$	IRKS $p = 4$
71	0.14	$8.33 \cdot 10^{+74}$	$1.75 \cdot 10^{+97}$	311	0.32	$1.02 \cdot 10^{+11}$	$1.06 \cdot 10^{+128}$
81	0.13	$5.59 \cdot 10^{+54}$	$4.42 \cdot 10^{+82}$	321	0.31	$3.68 \cdot 10^{-16}$	$2.85 \cdot 10^{+95}$
91	0.11	$3.70 \cdot 10^{+26}$	$1.28 \cdot 10^{+60}$	331	0.30	$1.02 \cdot 10^{-35}$	$8.50 \cdot 10^{+59}$
101	0.10	$2.92 \cdot 10^{-04}$	$3.48 \cdot 10^{+30}$	341	0.29	$1.08 \cdot 10^{-45}$	$1.08 \cdot 10^{+21}$
111	0.09	$2.83 \cdot 10^{-18}$	$2.97 \cdot 10^{-09}$	351	0.28	$4.14 \cdot 10^{-46}$	$1.47 \cdot 10^{-22}$

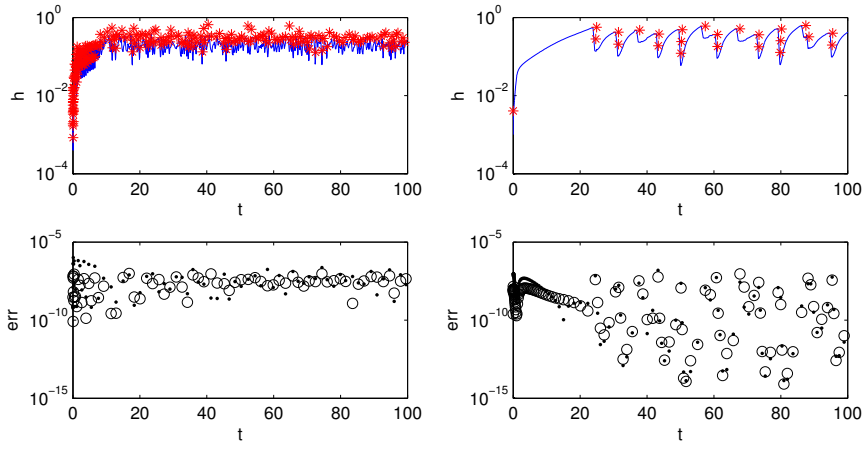

Figure 1. Results of method of order $p = 3$ with tolerance $tol = 10^{-6}$ on problem (4.2). Top: Rejected steps (*) and stepsize pattern (solid line); bottom error estimations (·) vs local error (∘). Left: standard step changing strategy, right: PI controller. Error figures - every 5-th point.

4.3 Variable stepsize implementation

In this section we show numerical results obtained by optimal methods found in [4] with $p = q = s = r - 1$, in a variable stepsize implementation. We adopted the rescale strategy described in Sec. 2.3 and we estimate the local error as described in Sec. 2.2.

As regards the estimation of the local error, we employ the error estimate described in Sec. 2.2 and compare it with the error computed as difference between the numerical solution and the exact one (when available), or between the numerical solution and the reference solution computed by applying the Matlab routine `ode15s` with the maximum precision (when the exact solution is not available).

Concerning the selection of the stepsize, we apply both strategies described

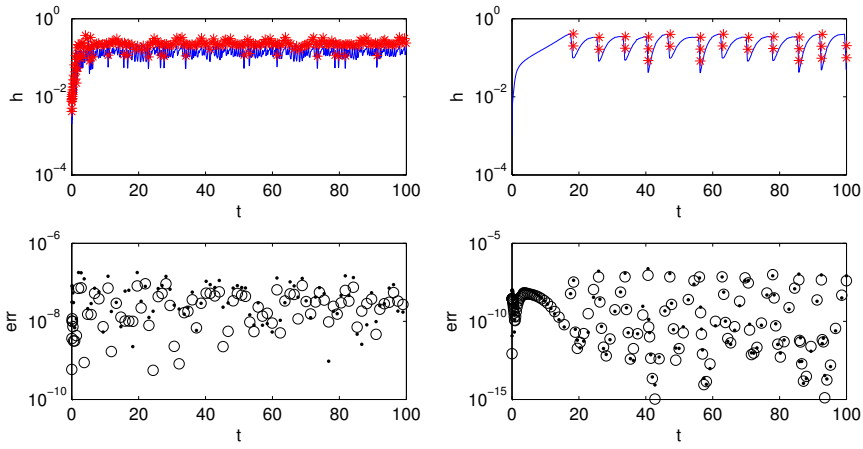

Figure 2. Results of method of order $p = 4$ with tolerance $tol = 10^{-6}$ on problem (4.2). Top: Rejected steps (*) and stepsize pattern (solid line); bottom error estimations (·) vs local error (∘). Left: standard step changing strategy, right: PI controller. Error figures - every 5-th point.

in Sec. 3.3 and compare them in terms of their efficiency. Such comparison takes into account the number of rejected steps with respect to those accepted and the smoothness of the patter of the stepsize on the overall integration interval, helpful to test the step changing strategy also taking into account the stability properties of the method. In fact, the more the pattern is smooth, the more the values of the stepsize lie within the stability region and, consequently, the more the number of rejections of the stepsize is small.

Fig. 1 and Fig. 2 report the results originated by applying the methods of order respectively 3 and 4 to (4.2), by employing both the standard and PI stepsize changing strategy. We observe that, in both cases, the error estimate is coherent with the true local error. Moreover, by employing the PI stepsize control, we observe a lower number of rejected steps which provides a lower computational cost: this advantage is commonly acknowledged in the literature (compare [17] and [14], where the variable stepsize implementation of two-step Runge-Kutta methods [12] based on modified collocation techniques [13] is presented).

Tab. 3–6 show the results achieved from the application of methods order from 1 up to 5 in variable stepsize environment, with several values of the tolerances (tol), for the numerical solution on (4.2) and (4.3). We compare global error (ge), total number of steps (ns), number of rejected steps (nrs), and total number of function evaluations (nfe). We observe that for more demanding tolerances high order methods have to be preferred. Moreover, in any reported case we observe the superiority of the PI stepsize controller in gaining the solution with a lower computational effort.
Finally, Tab. 7 and Tab. 8 provide a comparison between our method of order 4 (named NORD4 method) with the embedded pairs of continuous explicit

Table 3. Results of numerical experiments for Prothero-Robinson problem (4.2) on $t \in [0, 100]$ with standard stepsize changing strategy

	$p = 1$				$p = 2$			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$2.63E{-3}$	474	129	602	$7.3E{-4}$	424	113	1072
10^{-04}	$4.43E{-6}$	790	111	900	$2.3E{-5}$	487	112	1196
10^{-06}	$6.83E{-8}$	4671	92	4762	$6.7E{-7}$	813	104	1832
10^{-08}	$2.18E{-9}$	43565	80	43644	$7.7E{-9}$	2316	105	4840
10^{-10}				$> 10^5$	$3.9E{-11}$	9372	94	18930
10^{-12}				$> 10^5$	$3.0E{-14}$	42233	89	84642
	$p = 3$				$p = 4$			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$1.84E{-5}$	563	160	2166	$8.5E{-5}$	781	203	3932
10^{-04}	$4.60E{-7}$	593	170	2286	$7.6E{-6}$	780	200	3916
10^{-06}	$5.23E{-8}$	704	231	2802	$3.6E{-9}$	859	223	4324
10^{-08}	$3.07E{-10}$	1090	416	4515	$4.1E{-10}$	1020	267	5144
10^{-10}	$2.92E{-12}$	2270	870	9417	$4.3E{-12}$	1446	366	7244
10^{-12}	$7.03E{-14}$	5997	2463	25377	$4.4E{-14}$	2593	615	12828
	$p = 5$							
tol	ge	ns	nrs	nfe				
10^{-02}	$1.02E{-4}$	779	190	4840				
10^{-04}	$4.42E{-6}$	817	219	5175				
10^{-06}	$4.05E{-8}$	884	264	5735				
10^{-08}	$1.35E{-10}$	1037	383	7095				
10^{-10}	$5.54E{-13}$	1418	677	10470				
10^{-12}	$1.88E{-14}$	2047	926	14860				

Table 4. Results of numerical experiments for Prothero-Robinson problem (4.2) with PI controller, whose parameters are those defined in (3.3)

	$p=1$				$p=2$			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$9.67E{-5}$	566	20	585	$1.16E{-7}$	498	26	1046
10^{-04}	$3.97E{-8}$	1016	21	1036	$4.74E{-5}$	564	23	1172
10^{-06}	$6.20E{-9}$	4950	24	4973	$1.62E{-10}$	952	25	1952
10^{-08}	$1.84E{-10}$	43928	27	43954	$4.04E{-10}$	2512	25	5072
10^{-10}				$> 10^5$	$1.68E{-13}$	9620	25	19288
10^{-12}				$> 10^5$	$3.25E{-13}$	42523	26	85096
	$p=3$				$p=4$			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$2.71E{-4}$	397	5	1203	$9.86E{-6}$	531	20	2200
10^{-04}	$6.18E{-6}$	497	19	1545	$8.45E{-11}$	564	20	2332
10^{-06}	$1.38E{-12}$	622	22	1929	$1.01E{-8}$	612	17	2512
10^{-08}	$2.71E{-10}$	850	5	2562	$6.38E{-11}$	780	15	3176
10^{-10}	$2.71E{-12}$	1843	7	5547	$2.79E{-12}$	1119	20	4552
10^{-12}	$2.71E{-14}$	4896	14	14727	$3.12E{-16}$	1887	14	7600
	$p=5$							
tol	ge	ns	nrs	nfe				
10^{-02}	$1.66E{-5}$	584	19	3010				
10^{-04}	$1.53E{-8}$	582	14	2975				
10^{-06}	$3.76E{-11}$	643	17	3295				
10^{-08}	$5.80E{-12}$	715	12	3630				
10^{-10}	$1.22E{-13}$	965	19	4915				
10^{-12}	$2.28E{-20}$	1403	23	7125				

Runge-Kutta methods of order 3 and 4 (named CERK43 method, see [2]). It results that NORD4 method is able to achieve the accuracy of CERK43 with a lower computational cost. The requested computational effort is even lower when the variable stepsize strategy is based on the PI controller.

5 Concluding remarks

We have treated the implementation of explicit GLMs in Nordsieck form with quadratic stability and large stability regions, constructed in [4]. We addressed the issues related to the implementation in a variable step algorithm, in particular: local error estimation, computation of the input vector for the new stepsize, stepsize changing, starting procedure. Numerical experiments carried on significative test examples proved that:

- In a fixed stepsize algorithm, our methods converge with a larger stepsize with respect to IRKS methods.

Table 5. Results of numerical experiments for van der Pohl equation (4.3) with standard stepsize changing strategy

	$p=1$				$p=2$			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$4.09E{-4}$	3189	842	4030	$8.2E{-4}$	3071	889	7918
10^{-04}	$8.98E{-6}$	3207	863	4069	$2.0E{-6}$	2983	1184	8332
10^{-06}	$1.07E{-6}$	3356	861	4216	$4.8E{-9}$	3041	885	7850
10^{-08}	$1.06E{-6}$	4572	842	5413	$1.0E{-9}$	3185	919	8206
10^{-10}	$3.90E{-7}$	21316	4	21319	$3.E{-11}$	3747	892	9276
10^{-12}	$3.90E{-8}$	213112	4	213115	$3.E{-11}$	6534	864	14794
	$p=3$				$p=4$			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$6.84E{-4}$	3998	1193	15570	$1.1E{-3}$	5507	1366	27488
10^{-04}	$5.04E{-6}$	4037	1214	15750	$2.8E{-6}$	5532	1415	27784
10^{-06}	$9.01E{-9}$	4060	1225	15852	$7.7E{-9}$	5432	1392	27292
10^{-08}	$1.1E{-10}$	4108	1236	16029	$1.7E{-9}$	5527	1404	27720
10^{-10}	$6.8E{-13}$	4285	1335	16857	$2.E{-12}$	5640	1436	28300
10^{-12}	$6.8E{-13}$	4782	1548	18987	$7.E{-13}$	5826	1439	29056
	$p=5$							
tol	ge	ns	nrs	nfe				
10^{-02}	$2.17E{-4}$	5595	1400	34970				
10^{-04}	$3.65E{-7}$	5555	1365	34595				
10^{-06}	$7.52E{-9}$	5535	1375	34545				
10^{-08}	$3.3E{-10}$	5572	1394	34825				
10^{-10}	$1.5E{-11}$	5673	1458	35650				
10^{-12}	$1.3E{-12}$	5903	1637	37695				

Table 6. Results of numerical experiments for van der Pohl equation (4.3) with PI controller, whose parameters are those defined in (3.3)

	$p=1$				$p=2$			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$1.59E{-4}$	3732	166	3897	$6.42E{-3}$	2853	175	6054
10^{-04}	$1.04E{-5}$	3735	166	3900	$7.00E{-7}$	2889	178	6132
10^{-06}	$8.56E{-7}$	3934	169	4102	$4.51E{-11}$	2937	178	6228
10^{-08}	$8.37E{-7}$	5309	167	5475	$4.32E{-11}$	3117	183	6598
10^{-10}	$3.89E{-7}$	21564	12	21575	$3.41E{-11}$	3825	177	8002
10^{-12}	$3.90E{-8}$	213392	15	213406	$2.53E{-11}$	6807	140	13892
	$p=3$				$p=4$			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$2.69E{-4}$	2512	63	7722	$1.33E{-6}$	3342	139	13920
10^{-04}	$2.69E{-6}$	2297	9	6915	$1.68E{-9}$	3401	149	14196
10^{-06}	$2.69E{-8}$	2314	5	6954	$7.47E{-9}$	3480	158	14548
10^{-08}	$2.7E{-10}$	2359	4	7086	$8.70E{-11}$	3519	155	14692
10^{-10}	$2.7E{-12}$	2536	7	7626	$5.93E{-12}$	3578	151	14912
10^{-12}	$6.8E{-13}$	3214	58	9813	$6.95E{-13}$	3882	149	15880
	$p=5$							
tol	ge	ns	nrs	nfe				
10^{-02}	$2.03E{-6}$	3669	124	18960				
10^{-04}	$7.29E{-9}$	3785	145	19645				
10^{-06}	$2.21E{-8}$	3727	127	19265				
10^{-08}	$1.22E{-9}$	3871	147	20085				
10^{-10}	$7.2E{-11}$	4055	150	21020				
10^{-12}	$5.0E{-12}$	4436	133	22840				

Table 7. Numerical comparison of CERK43 and NORD4 methods applied on the Prothero-Robinson equation (4.2), with standard stepsize control

	CERK43				NORD4			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$7.25 \cdot 10^{-5}$	734	437	5852	$8.53 \cdot 10^{-5}$	781	203	3932
10^{-04}	$1.62 \cdot 10^{-6}$	758	408	5827	$7.63 \cdot 10^{-6}$	780	200	3916
10^{-06}	$1.11 \cdot 10^{-8}$	904	396	6497	$3.63 \cdot 10^{-9}$	859	223	4324
10^{-08}	$1.15 \cdot 10^{-10}$	1415	370	8922	$4.1 \cdot 10^{-10}$	1020	267	5144
10^{-10}	$5.77 \cdot 10^{-12}$	3092	353	17222	$4.3 \cdot 10^{-12}$	1446	366	7244
10^{-12}	$4.23 \cdot 10^{-6}$	8451	337	43937	$4.4 \cdot 10^{-14}$	2593	615	12828

Table 8. Numerical comparison of CERK43 and NORD4 methods applied on the Van der Pohl oscillator (4.3), with standard stepsize control

	CERK43				NORD4			
tol	ge	ns	nrs	nfe	ge	ns	nrs	nfe
10^{-02}	$2.32 \cdot 10^{-4}$	5216	3130	41727	$1.07E{-3}$	5507	1366	27488
10^{-04}	$4.93 \cdot 10^{-5}$	5220	3129	41742	$2.83E{-6}$	5532	1415	27784
10^{-06}	$4.16 \cdot 10^{-6}$	5226	3127	41762	$7.70E{-9}$	5432	1392	27292
10^{-08}	$2.50 \cdot 10^{-8}$	5251	3127	41887	$1.67E{-9}$	5527	1404	27720
10^{-10}	$2.56 \cdot 10^{-8}$	5327	3125	42257	$1.52E{-12}$	5640	1436	28300
10^{-12}	$7.67 \cdot 10^{-9}$	6101	4444	52722	$7.08E{-13}$	5826	1439	29056

- In a variable stepsize algorithm, our methods are more efficient than other existing methods.
- PI controller strategy considerably reduces the number of rejected steps, and therefore improves the efficiency of our methods.

The next step of our research will be the development of a variable step variable order algorithm. The design of a reliable order changing strategy is based on a fundamental building block which is provided by an accurate estimation of the higher order terms in the local error expansion. We aim to derive such an estimate by suitably extending the results reported in [8].

Acknowledgements

Most of this work was done during the visit of the first author at Università degli Studi di Salerno in March 2012. This author would like to thank Dipartimento di Matematica for the hospitality during the visit.

References

[1] Z. Bartoszewski and Z. Jackiewicz. Nordsieck representation of two-step Runge-Kutta methods for ordinary differential equations. *Appl. Numer. Math.*, **53**(2-4):149–163, 2005.

[2] Z. Bartoszewski and Z. Jackiewicz. Explicit Nordsieck methods with extended stability regions. *Appl. Math. Comput.*, **218**(10):6056–6066, 2012.

[3] M. Braś. Nordsieck methods with inherent quadratic stability. *Math. Model. Anal.*, **16**(1):82–96, 2011.

[4] M. Braś and A. Cardone. Construction of efficient general linear methods for non-stiff differential systems. *Math. Model. Anal.*, **17**(2):171–189, 2012.

[5] J. C. Butcher. *Numerical methods for ordinary differential equations*. John Wiley & Sons Ltd., Chichester, 2003.

[6] J. C. Butcher, P. Chartier and Z. Jackiewicz. Nordsieck representation of DIMSIM. *Numer. Algorithms*, **16**:209–230, 1997.

[7] J. C. Butcher and Z. Jackiewicz. Construction of general linear methods with Runge-Kutta stability properties. *Numer. Algorithms*, **36**(1):53–72, 2004.

[8] J. C. Butcher, Z. Jackiewicz and W. M. Wright. Error propagation of general linear methods for ordinary differential equations. *J. Complexity*, **23**(4-6):560–580, 2007.

[9] A. Cardone and Z. Jackiewicz. Explicit Nordsieck methods with quadratic stability. *Numer. Algorithms*, **60**:1–25, 2012.

[10] A. Cardone, Z. Jackiewicz and H.D. Mittelmann. Optimization-based search for Nordsieck methods of high order with quadratic stability. *Math. Model. Anal.*, **17**(3):293–308, 2012.

[11] D. Conte, R. D'Ambrosio and Z. Jackiewicz. Two-step Runge-Kutta methods with quadratic stability functions. *J. Sci. Comput.*, **44**(2):191–218, 2010.

[12] D. Conte, R. D'Ambrosio, Z. Jackiewicz and B. Paternoster. A pratical approach for the derivation of algebraically stable two-step Runge-Kutta methods. *Math. Model. Anal.*, **17**(1):65–277, 2012.

[13] R. D'Ambrosio, M. Ferro, Z. Jackiewicz and Paternoster B. Two-step almost collocation for ordinary differential equations. *Numer. Algorithms*, **53**(2-3):195–217, 2010.

[14] R. D'Ambrosio and Z. Jackiewicz. Construction and implementation of highly stable two-step continuous methods for stiff differential systems. *Math. Comput. Simul.*, **81**(9):1707–1728, 2011.

[15] K. Gustafsson, M. Lundh and G. Söderlind. A *PI* stepsize control for the numerical solution of ordinary differential equations. *BIT*, **28**(2):270–387, 1988.

[16] E. Hairer, S. P. Nørsett and G. Wanner. *Solving Ordinary Differential Equations I. Nonstiff Problems*. Springer-Verlag, Berlin, Heidelberg, New York, 2000.

[17] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, Second Revised Edition*. Springer-Verlag, Berlin, Heidelberg, New York, 1996.

[18] S. J. Y. Huang. *Implementation of General Linear Methods for Stiff Ordinary Differential Equations*. The University of Auckland, New Zealand, 2005.

[19] Z. Jackiewicz. *General linear methods for ordinary differential equations*. John Wiley & Sons Inc., Hoboken, NJ, 2009.

[20] A. Prothero and A. Robinson. On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations. *Math. Comput*, **28**(125):145–162, 1974.

[21] W. M. Wright. *General linear methods with inherent Runge-Kutta stability*. Doctoral thesis, The University of Auckland, New Zealand, 2002.