

A CLASS OF NUMERICAL METHODS FOR AUTONOMOUS INITIAL VALUE PROBLEMS

FLAVIUS-OLIMPIU PĂTRULESCU*

Abstract. In this paper we introduce a class of explicit numerical methods for approximating the solutions of scalar initial value problems for first order differential equations, using a nonlinear interpolation formula. We show that the methods generated in this way can be identified as explicit Runge-Kutta methods and we analyze some particular cases. Finally, numerical examples are provided.

MSC 2000. 65L05, 65L06, 65L20.

Keywords. initial value problem, stability region, convergence order, local truncation error.

1. INTRODUCTION

Consider a scalar *initial value problem* (IVP):

$$(1) \quad \begin{cases} y' = f(x, y), & x \in I \\ y(x_0) = y_0, \end{cases}$$

where: $I \subseteq \mathbb{R}$, $y_0 \in \mathbb{R}$, $f : I \times \mathbb{R} \rightarrow \mathbb{R}$ and $x_0 \in I$. We assume that $I = [x_0, x_0 + T]$, $0 < T < \infty$ and the function f satisfies all requirements necessary to insure the existence of a unique solution y on the bounded interval I , see [1], [2] for details.

In this paper we present a numerical method to approximate the solution of IVP (1) using a particular case of a nonlinear interpolation formula analyzed in [5] and [7]. Another particular case was utilized in [4].

For a 5-times differentiable function $h : I \rightarrow \mathbb{R}$ consider the function $g : I \rightarrow \mathbb{R}$ given by

$$(2) \quad g(x) = h(x_0) + \frac{1}{2}(x-x_0)[h'(x_0 + \alpha_1(x-x_0)) + h'(x_0 + \alpha_2(x-x_0))], \quad x \in I,$$

where $\alpha_1 = \frac{3-\sqrt{3}}{6}$ and $\alpha_2 = \frac{3+\sqrt{3}}{6}$.

It is showed in [5] that the function g verifies

$$(3) \quad h^{(i)}(x_0) = g^{(i)}(x_0), \quad i = \overline{0, 4} \quad \text{and} \quad |h(x) - g(x)| \leq \frac{71}{4320} M_5 |x - x_0|^5, \quad x \in I,$$

This work has been supported in part by grant no. PN II RU TE 2011-3-0013 .

*Tiberiu Popoviciu Institute of Numerical Analysis, Romanian Academy, P.O. Box 68–1, 400110, Cluj-Napoca, Romania, e-mail: fpatrulescu@ictp.acad.ro.

where $M_5 = \sup_{x \in I} |h^{(5)}(x)|$. Also, we observe that the coefficients α_1, α_2 satisfy the equalities

$$(4) \quad \begin{cases} \alpha_1 + \alpha_2 = 1, & \alpha_1^2 + \alpha_2^2 = \frac{2}{3}, \\ \alpha_1^3 + \alpha_2^3 = \frac{1}{2}, & \alpha_1^4 + \alpha_2^4 = \frac{7}{18}. \end{cases}$$

In the next sections, for simplicity, we consider only the autonomous case, i.e. $f = f(y)$. The general case can be treated using similar arguments.

The paper is structured as follows. In Section 2 we derive the numerical method and in Section 3 we analyze its versions corresponding to the particular cases of the parameter. In Section 4 numerical examples are provided to exemplify the results obtained in the previous sections.

2. THE NUMERICAL METHOD

The interval $I = [x_0, x_0 + T]$ is partitioned by the point set $\{x_n | n = \overline{0, N}, N \in \mathbb{N}\}$, where x_n are given by a prescribed rule, and we denote by y_n a numerical approximation to the exact solution y of (1) at x_n .

We suppose that the exact solution y of initial value problem (1) is 5-times differentiable at least. Conditions for regularity of exact solution of an initial value problem can be found in [2].

Using the results presented above we deduce that there exists an approximation \tilde{y} of y given by

$$(5) \quad \begin{aligned} \tilde{y}(x) &= y(x_0) + \frac{1}{2}(x - x_0)[y'(x_0 + \alpha_1(x - x_0)) + y'(x_0 + \alpha_2(x - x_0))] \\ &= y(x_0) + \frac{1}{2}(x - x_0)[f(y(x_0 + \alpha_1(x - x_0))) + f(y(x_0 + \alpha_2(x - x_0)))], \end{aligned}$$

for all $x \in I$. From (3) we deduce that this approximation verifies

$$\tilde{y}^{(i)}(x_0) = y^{(i)}(x_0), \quad i = \overline{0, 4} \text{ and } |\tilde{y}(x) - y(x)| < \frac{71}{4320} M_5 |x - x_0|^5, \quad x \in I,$$

where $M_5 = \sup_{x \in I} |y^{(5)}(x)|$.

The unknown quantities $y(x_0 + \alpha_i(x - x_0))$, $i = 1, 2$, in (5) can be approximated in the same manner and we continue this approximation procedure for the next unknown values $y(x_0 + \alpha_1^p \alpha_2^q(x - x_0))$, $p, q = 1, 2, \dots$.

Using the notation

$$u_{rs}(x) = y(x_0 + \alpha_1^r \alpha_2^s(x - x_0))$$

the algorithm described above can be written for the firsts p steps in the following compact form

$$(6) \quad \begin{aligned} u_{00}(x) &= y(x_0) + \frac{1}{2}(x - x_0)[f(u_{10}(x)) + f(u_{01}(x))] \\ u_{i-jj}(x) &= y(x_0) + \frac{1}{2} \alpha_1^{i-j} \alpha_2^j (x - x_0) [f(u_{i-j+1j}(x)) + f(u_{i-jj+1}(x))] \\ &\quad j = 0, \dots, i, \quad i = 1, \dots, p \end{aligned}$$

and we can define $\tilde{y}(x) = u_{00}(x)$, $x \in I$.

Taking into account that

$$\frac{1}{2}\alpha_1^{p-j}\alpha_2^j(x-x_0) \rightarrow 0, \quad j = \overline{0, p}, \quad \text{as } p \rightarrow \infty,$$

we can choose $p = p_0$ such that $y(x_0)$ approximates $u_{p_0 0}, u_{p_0-10}, \dots, u_{0p_0}$ with any given accuracy, since y is a continuous function. From (6) we obtain

$$\begin{aligned} (7) \quad \tilde{u}_{00}(x) &= y(x_0) + \frac{1}{2}(x-x_0)[f(\tilde{u}_{10}(x)) + f(\tilde{u}_{01}(x))] \\ \tilde{u}_{i-jj}(x) &= y(x_0) + \frac{1}{2}\alpha_1^{i-j}\alpha_2^j(x-x_0)[f(\tilde{u}_{i-j+1j}(x)) + f(\tilde{u}_{i-jj+1}(x))] \\ &\quad j = 0, \dots, i, \quad i = \overline{1, \dots, p_0-1} \\ \tilde{u}_{p_0-jj}(x) &= y(x_0), \quad j = \overline{0, \dots, p_0}. \end{aligned}$$

We use this recurrence relation to construct a numerical method for the scalar initial value problem (1). Replacing x by x_1 in (7) we obtain an approximation $\tilde{u}_{00}(x_1)$ for the exact value $y(x_1)$. We denote this approximation by y_1 and we apply again the algorithm (7) for x_2 , but, instead of $y(x_0) = y_0$, we consider the value y_1 previously computed. We repeat this procedure for each x_n , $n = \overline{1, N}$.

Using the notation

$$u_{qr}^n = y_n + \frac{1}{2}\alpha_1^q\alpha_2^r h_n [f(u_{q+1r}^n) + f(u_{qr+1}^n)],$$

the above algorithm can be written in the following way

$$\begin{aligned} (8) \quad y_{n+1} &= y_n + \frac{1}{2}h_n [f(u_{10}^n) + f(u_{01}^n)] \\ u_{i-jj}^n &= y_n + \frac{1}{2}\alpha_1^{i-j}\alpha_2^j h_n [f(u_{i-j+1j}^n) + f(u_{i-jj+1}^n)] \\ &\quad j = \overline{0, i}, \quad i = \overline{1, p_0-1}, \\ u_{p_0-jj}^n &= y_n, \quad j = \overline{0, p_0} \end{aligned}$$

where $h_n = x_{n+1} - x_n$, $n = \overline{0, N-1}$.

For $p_0 = 1$ we obtain the *Euler forward method* given by

$$(9) \quad y_{n+1} = y_n + h_n f(y_n), \quad n = \overline{1, N-1}.$$

We have the following equivalence result.

THEOREM 1. *The method (8) can be identified as an $\frac{p_0(p_0+1)}{2}$ -stages explicit Runge-Kutta method with the Butcher array given by*

$$c = [0, \alpha_1^{p_0-1}, \alpha_1^{p_0-2}\alpha_2, \dots, \alpha_2^{p_0-1}, \dots, \alpha_1^{p_0-j}, \alpha_1^{p_0-j-1}\alpha_2, \dots, \alpha_2^{p_0-j}, \dots, \alpha_1, \alpha_2]^T, \\ b^T = [0, 0, \dots, 0, \frac{1}{2}, \frac{1}{2}]$$

$$A = \begin{bmatrix} B_{p_0-1} & \ddots & & & \\ \ddots & B_{p_0-2} & \ddots & \circ & \\ & \ddots & B_{p_0-3} & \ddots & \\ & & \circ & \ddots & \ddots & \ddots \\ & & & & \ddots & B_1 \end{bmatrix}$$

where

$$B_{p_0-1} = \begin{bmatrix} 0 \\ \alpha_1^{p_0-1} \\ \alpha_1^{p_0-2} \alpha_2 \\ \dots \\ \alpha_2^{p_0-1} \end{bmatrix} \circ,$$

$$B_{p_0-j} = \begin{bmatrix} \frac{\alpha_1^{p_0-j}}{2} & \frac{\alpha_1^{p_0-j}}{2} & & & \\ & \frac{\alpha_1^{p_0-j-1} \alpha_2}{2} & \frac{\alpha_1^{p_0-j-1} \alpha_2}{2} & & \\ & \circ & \dots & \circ & \\ & & \frac{\alpha_2^{p_0-j}}{2} & \frac{\alpha_2^{p_0-j}}{2} & \end{bmatrix}, \quad j = \overline{2, p_0 - 1}.$$

Proof. We know that for a q -stages explicit Runge-Kutta method with the Butcher array (see e.g. [1], [3])

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array}$$

the approximation for the new point $x_{n+1} = x_n + h_n$ is given by

$$(10) \quad y_{n+1} = y_n + h_n \sum_{i=1}^q b_i k_i,$$

where

$$\begin{cases} k_1 = f(x_n, y_n), \\ k_i = f(x_n + c_i h_n, y_n + h_n \sum_{j=1}^{i-1} a_{ij} k_j), \quad i = \overline{2, q}. \end{cases}$$

It easy to see that (10) provides for autonomous case and for the above Butcher array a rule of the form (8), which concludes the proof. \square

3. METHOD (??) WHEN $p_0 = 2, 3$ AND 4

For $p_0 = 2$ we obtain

$$(11) \quad y_{n+1} = y_n + \frac{1}{2} h_n [f(y_n + \alpha_1 h_n f(y_n)) + f(y_n + \alpha_2 h_n f(y_n))], \quad n = \overline{1, N-1}.$$

The method (11) can be written in a more suitable form

$$y_{n+1} = y_n + \frac{1}{2} h_n [f(u_{10}^n) + f(u_{01}^n)],$$

$$u_{10}^n = y_n + \alpha_1 h_n f(y_n), \quad u_{01}^n = y_n + \alpha_2 h_n f(y_n)$$

and can be identified as a 3-stages explicit Runge-Kutta method with the Butcher array given by

$$\begin{array}{c|ccc} & 0 & & \\ \alpha_1 & \alpha_1 & 0 & \\ \alpha_2 & \alpha_2 & 0 & 0 \\ \hline & 0 & \frac{1}{2} & \frac{1}{2} \end{array}$$

As in [6], we suppose that

$$(12) \quad \|f\| < M \text{ and } \|f^{(j)}\| < \frac{L^j}{M^{j-1}},$$

where $\|f\| = \sup\{|f(t)| : t \in I\}$ and M, L are positive real numbers.

The convergence order of the method (11) is provided in the following result.

THEOREM 2. *The method (11) has convergence order 2 and the coefficient of principal local truncation error C_3 has the following bound $\|C_3\| \leq \frac{1}{6}ML^2$.*

Proof. Following [3], in order to obtain the local truncation error of the method (11) we consider the operator

$$(13) \quad L[z(x); h] = z(x+h) - z(x) - \frac{h}{2}[f(z(x) + \alpha_1 f(z(x))) + f(z(x) + \alpha_2 f(z(x)))],$$

where z is an arbitrary function defined on I , sufficiently differentiable and $z'(x) = f(z(x))$, for all $x \in I$.

Expanding in Taylor series with respect to x and using (4) we obtain

$$(14) \quad L[z(x); h] = \frac{h^3}{6}[f'(z(x))]^2 f(z(x)) + O(h^4).$$

Then, using the definition for the convergence order given in [3] we deduce that the method has second-order of accuracy. Also, substituting z by the exact solution y , x by x_n and supposing *localizing assumption* $y_i = y(x_i)$, $i = \overline{1, n}$ the *local truncation error* can be written as

$$(15) \quad T_{n+1} = \frac{h^3}{6}[f'(y(x_n))]^2 f(y(x_n)) + O(h^4).$$

Moreover, the coefficient of *principal local truncation error* is given by

$$(16) \quad C_3 = \frac{1}{6}[f'(y(x_n))]^2 f(y(x_n)).$$

Next, using (12) we have for C_3 the following bound

$$\|C_3\| = \frac{1}{6}\|(f')^2 f\| \leq \frac{1}{6}\|f'\|^2 \|f\| \leq \frac{1}{6}ML^2,$$

which concludes the proof. \square

Following [2], for the Runge-Kutta methods in addition of the *row-sum condition*, $Ae = Ce$, and *consistency condition*, $b^T e = 1$, the sufficient conditions for order 2 are given by

$$(17) \quad b^T C e = \frac{1}{2} \quad b^T A e = \frac{1}{2},$$

where $e = [1, \dots, 1]$ and $C = \text{diag}(c)$. A simple calculus shows that the method (11) verifies conditions (17) and it represents a validation for (14).

The *stability function* and *absolute stability region* of the method (11) are provided in the following result.

THEOREM 3. *The method (11) has the stability function given by*

$$(18) \quad R(z) = 1 + z + \frac{z^2}{2}, \quad z \in \mathbb{C}.$$

Proof. Following [3], we apply the method (11) to the scalar test equation

$$y' = \lambda y, \quad \lambda \in \mathbb{C}, \quad \operatorname{Re} \lambda < 0,$$

and we obtain the difference equation

$$y_{n+1} = y_n [1 + (\lambda h_n) + \frac{1}{2}(\alpha_1 + \alpha_2)(\lambda h_n)^2].$$

Denoting $z = \lambda h_n$ and using (4) we obtain the stability function

$$R(z) = 1 + z + \frac{z^2}{2}.$$

Note that this function is the same as the stability function for 2 stages explicit Runge-Kutta methods of order 2. \square

The absolute stability region is given by

$$(19) \quad \mathcal{R} = \{z \in \mathbb{C} : |1 + z + \frac{1}{2}z^2| < 1\}$$

and it is plotted using scanning technique in Figure 1. The computations have been carried out using Matlab software.

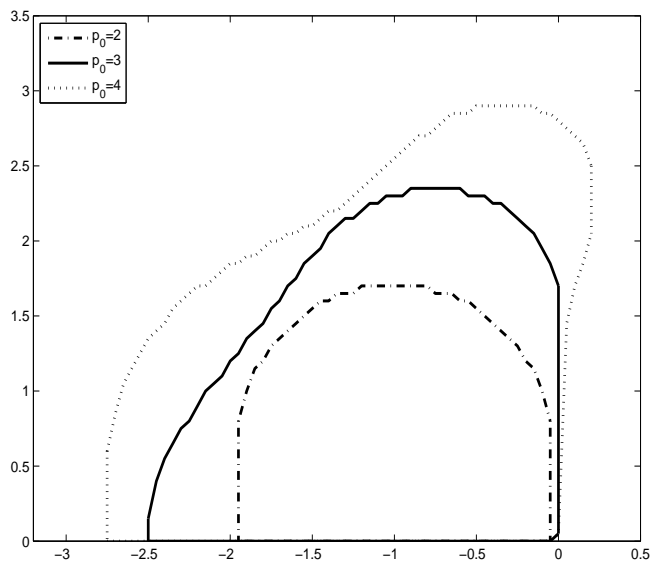


Fig. 1. Absolute stability regions for method (8) when $p_0 = 2, 3, 4$.

For $p_0 = 3$ we obtain the method

$$(20) \quad \begin{aligned} y_{n+1} &= y_n + \frac{1}{2}h_n[f(u_{10}^n) + f(u_{01}^n)] \\ u_{10}^n &= y_n + \frac{1}{2}\alpha_1 h_n[f(u_{20}^n) + f(u_{11}^n)] \\ u_{01}^n &= y_n + \frac{1}{2}\alpha_2 h_n[f(u_{11}^n) + f(u_{02}^n)] \\ u_{2-jj}^n &= y_n + \alpha_1^{2-j}\alpha_2^j h_n f(y_n), \quad j = \overline{0, 2} \end{aligned}$$

which can be identified as a 6–stages explicit Runge-Kutta method with the Butcher array given by

	α_1^2	0	0	0	0	0
	$\alpha_1\alpha_2$	$\alpha_1\alpha_2$	0	0	0	0
	α_2^2	α_2^2	0	0	0	0
	α_1	0	$\frac{\alpha_1}{2}$	$\frac{\alpha_1}{2}$	0	0
	α_2	0	0	$\frac{\alpha_2}{2}$	$\frac{\alpha_2}{2}$	0
		0	0	0	0	$\frac{1}{2}$
		0	0	0	0	$\frac{1}{2}$

The convergence order of the method (20) is provided in the following result.

THEOREM 4. *The method (20) has convergence order 3 and the coefficient of the principal local truncation error C_4 has the following bound $\|C_4\| \leq \frac{1}{24}ML^3$.*

Proof. Using (4) and similar arguments to those presented in the proof of Theorem 2 we obtain

$$(21) \quad L[z(x); h] = \frac{h^4}{24}[f'(z(x))]^3 f(z(x)) + O(h^5).$$

Therefore we deduce that this method has convergence order 3. Also, we can determine the *local truncation error* as

$$(22) \quad T_{n+1} = \frac{h^4}{24}[f'(y(x_n))]^3 f(y(x_n)) + O(h^5)$$

and the coefficient of principal local truncation error is given by

$$C_4 = \frac{1}{24}[f'(y(x_n))]^3 f(y(x_n)).$$

Using (12) we obtain $\|C_4\| < \frac{1}{24}ML^3$, which concludes the proof. \square

A simple calculus shows that the method (20) verifies the sufficient conditions for order 3, see [2] for details,

$$(23) \quad b^T C e = \frac{1}{2}, \quad b^T C^2 e = \frac{1}{3}, \quad b^T A C e = \frac{1}{6}$$

and it represents a validation for (21).

THEOREM 5. *The method (20) has the stability function given by*

$$(24) \quad R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6}, \quad z \in \mathbb{C}.$$

Proof. Using similar arguments as in the proof of Theorem 3. \square

We observe that method (20) has the same stability function as 3 stages explicit Runge-Kutta methods of order 3. The absolute stability region defined by

$$(25) \quad \mathcal{R} = \{z \in \mathbb{C} : |1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3| < 1\}$$

is plotted in Figure 1.

For $p_0 = 4$ we obtain the method

$$(26) \quad \begin{aligned} y_{n+1} &= y_n + \frac{1}{2}h_n[f(u_{10}^n) + f(u_{01}^n)] \\ u_{i-jj}^n &= y_n + \frac{1}{2}\alpha_1^{i-j}\alpha_2^j h_n[f(u_{i-j+1j}^n) + f(u_{i-jj+1}^n)] \\ &\quad j = \overline{0, i}, \quad i = \overline{1, 2}, \\ u_{3-jj}^n &= y_n + \alpha_1^{3-j}\alpha_2^j h_n f(y_n) \quad j = \overline{0, 3} \end{aligned}$$

and can be identified as a 10-stages explicit Runge-Kutta method with the Butcher array given by

0	0									
α_1^3	α_1^3	0								
$\alpha_1^2\alpha_2$	$\alpha_1^2\alpha_2$	0	0							
$\alpha_1\alpha_2^2$	$\alpha_1\alpha_2^2$	0	0	0						
α_2^3	α_2^3	0	0	0	0					
α_1^2	0	$\frac{\alpha_1^2}{2}$	$\frac{\alpha_1^2}{2}$	0	0	0				
$\alpha_1\alpha_2$	0	0	$\frac{\alpha_1\alpha_2}{2}$	$\frac{\alpha_1\alpha_2}{2}$	0	0	0			
α_2^2	0	0	0	$\frac{\alpha_2^2}{2}$	$\frac{\alpha_2^2}{2}$	0	0	0		
α_1	0	0	0	0	0	$\frac{\alpha_1}{2}$	$\frac{\alpha_1}{2}$	0	0	
α_2	0	0	0	0	0	0	$\frac{\alpha_2}{2}$	$\frac{\alpha_2}{2}$	0	0
	0	0	0	0	0	0	0	0	$\frac{1}{2}$	$\frac{1}{2}$

Using (4) and similar arguments to those used above we deduce that the method (26) has convergence order 4. The stability function is given by

$$(27) \quad R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24}$$

and we observe that it is the same as stability function for 4 stages explicit Runge-Kutta methods of order 4. The absolute stability region defined by

$$(28) \quad \mathcal{R} = \{z \in \mathbb{C} : |1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3 + \frac{1}{24}z^4| < 1\}$$

is plotted in Figure 1.

We restrict ourselves to the cases when p_0 takes the values 2, 3 and 4. For $p_0 \geq 5$ methods obtain from (8) have a form much more complicated with a high cost of calculus and the results concerning accuracy are not so outstanding. From the next theorem we deduce that the convergence order is not greater than 4.

THEOREM 6. *For $p_0 \geq 5$ the method (8) has maximal convergence order 4.*

Proof. Following [3], we consider the operator

$$\begin{aligned} L[z(x), h] &= z(x+h) - z(x) - \frac{h}{2}[f(z(x+\alpha_1 h)) + f(z(x+\alpha_2 h))] \\ &= z(x+h) - z(x) - \frac{h}{2}[z'(x+\alpha_1 h) + z'(x+\alpha_2 h)], \end{aligned}$$

where z is an arbitrary function defined on I , 5-times differentiable at least and $z'(x) = f(z(x))$, for all $x \in I$.

Next, using Taylor series with respect to x and (4) we obtain

$$L[z(x), h] = h^5 \left(\frac{1}{120} - \frac{1}{48} \frac{7}{18} \right) z^{(5)}(x) + O(h^6).$$

Then, from the definition of convergence order given in [3] we deduce that for $p_0 \geq 5$ method (8) has convergence order 4, which concludes the proof. \square

Note that the method (8) is a *zero-stable* method because it verifies *root-condition*. Also, since the convergence order is greater than 2 for all $p_0 \geq 2$ we conclude that it satisfies the *consistency condition*. It follows that our method represents a convergent method, see [3] for details.

4. NUMERICAL EXAMPLES

We consider the initial value problems.

EXAMPLE 7.

$$(29) \quad \begin{cases} y'(x) = \cos^2(y(x)), & x \in [0, 20] \\ y(0) = 0. \end{cases}$$

The exact solution is $y : [0, 20] \rightarrow \mathbb{R}$ given by $y(x) = \arctan(x)$. \square

EXAMPLE 8.

$$(30) \quad \begin{cases} y'(x) = \frac{y(x)}{4} \left(1 - \frac{y(x)}{20} \right), & x \in [0, 20] \\ y(0) = 1. \end{cases}$$

The exact solution is $y : [0, 20] \rightarrow \mathbb{R}$ given by

$$y(x) = \frac{20}{1+19 \exp(-x/4)}. \quad \square$$

We apply the methods (11), (20) and (26) with constant steplength to determine numerical solutions for the above examples.

As a measure of the performance we consider the errors obtained as the maximum of the absolute errors on the mesh points $x_n = nh$, $n = \overline{0, N}$

$$E_{\max} = \max\{|y(x_n) - y_n| : n = 0, 1, \dots, N\},$$

for different values of step h .

In the next tables we present results given by the proposed methods (11), (20) (26) in comparison with results given by Adams-Bashforth methods of orders 2, 3, 4 defined by

$$\begin{aligned} A - B(2) : \quad y_{n+1} - y_n &= \frac{h}{2}(3f_n - f_{n-1}), \\ A - B(3) : \quad y_{n+1} - y_n &= \frac{h}{12}(23f_n - 16f_{n-1} + 5f_{n-2}), \end{aligned}$$

$$A - B(4) : \quad y_{n+1} - y_n = \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3}).$$

Also, we present results given by the explicit Runge-Kutta methods with order equal with number of stages defined by the following Butcher arrays

$$\begin{array}{c}
 \begin{array}{c|c}
 0 & \\
 \frac{1}{2} & 0 \\
 \hline
 & 0 \quad 1
 \end{array} \\
 R - K(2)
 \end{array}
 \quad
 \begin{array}{c}
 \begin{array}{c|cc}
 0 & & \\
 \frac{1}{2} & \frac{1}{2} & \\
 1 & -1 & 2 \\
 \hline
 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6}
 \end{array} \\
 R - K(3)
 \end{array}
 \quad
 \begin{array}{c}
 \begin{array}{c|ccc}
 0 & & & \\
 \frac{1}{2} & \frac{1}{2} & & \\
 \frac{1}{2} & 0 & \frac{1}{2} & \\
 1 & 0 & 0 & 1 \\
 \hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
 \end{array} \\
 R - K(4)
 \end{array}$$

h	$1.0e - 01$	$1.0e - 02$	$1.0e - 03$	$1.0e - 04$	$1.0e - 05$
(11)	$5.755e - 04$	$5.415e - 06$	$5.381e - 08$	$5.378e - 10$	$5.377e - 12$
$A - B(2)$	$2.209e - 03$	$2.251e - 05$	$2.256e - 07$	$2.257e - 09$	$2.257e - 11$
$R - K(2)$	$5.755e - 04$	$5.415e - 06$	$5.381e - 08$	$5.378e - 10$	$5.377e - 12$
(20)	$1.333e - 05$	$1.244e - 08$	$1.235e - 11$	$1.254e - 14$	$2.464e - 14$
$A - B(3)$	$1.109e - 03$	$1.166e - 06$	$1.166e - 09$	$1.166e - 12$	$2.464e - 14$
$R - K(3)$	$2.028e - 05$	$2.077e - 08$	$2.082e - 11$	$1.976e - 14$	$2.486e - 14$
(26)	$2.202e - 07$	$2.050e - 11$	$2.886e - 15$	$1.110e - 14$	$2.464e - 14$
$A - B(4)$	$1.109e - 03$	$1.166e - 06$	$1.166e - 09$	$1.166e - 12$	$2.442e - 14$
$R - K(4)$	$5.357e - 07$	$5.337e - 11$	$5.884e - 15$	$1.132e - 14$	$2.486e - 14$

Table 1. E_{\max} for *Example 7*

h	$1.0e - 01$	$1.0e - 02$	$1.0e - 03$	$1.0e - 04$	$1.0e - 05$
(11)	$5.878e - 04$	$5.952e - 06$	$5.959e - 08$	$5.960e - 10$	$5.929e - 12$
$A - B(2)$	$1.892e - 03$	$1.907e - 05$	$1.908e - 07$	$1.908e - 09$	$1.892e - 11$
$R - K(2)$	$4.805e - 04$	$4.861e - 06$	$4.867e - 08$	$4.866e - 10$	$4.757e - 12$
(20)	$2.725e - 06$	$2.764e - 09$	$2.744e - 12$	$4.192e - 13$	$6.359e - 13$
$A - B(3)$	$1.387e - 03$	$1.404e - 05$	$1.406e - 07$	$1.406e - 09$	$1.404e - 11$
$R - K(3)$	$4.048e - 06$	$4.083e - 09$	$4.137e - 12$	$4.209e - 13$	$6.359e - 13$
(26)	$9.951e - 09$	$9.912e - 13$	$6.750e - 14$	$4.192e - 13$	$6.359e - 13$
$A - B(4)$	$1.420e - 03$	$1.407e - 05$	$1.406e - 07$	$1.406e - 09$	$1.404e - 11$
$R - K(4)$	$1.779e - 08$	$1.788e - 12$	$6.750e - 14$	$4.192e - 13$	$6.323e - 13$

Table 2. E_{\max} for *Example 8*

We analyze the results and we can see that for the above examples the proposed methods are at least comparable to the classical ones.

We observe that if length of the step decreases 10 times then the error magnitude for methods (11), (20) and (26) decreases 10^2 , 10^3 and 10^4 times, respectively. These results represent a validation of the fact that the convergence orders for the methods (11), (20) and (26) are 2, 3 and 4, respectively.

Because the method is explicit we indicate to be used especially for non-stiff problems, where the requirements on the steplength imposed by stability are no restrictive than the requirements imposed by accuracy, see [1] and [3] for

more details. Also, we expect that the method (8) gives better results in the case of a variable steplength.

REFERENCES

- [1] J.C. BUTCHER, *Numerical Methods for Ordinary Differential Equations*, John Wiley & Sons, 2008.
- [2] M. CROUZEIX and A.L. MIGNOT, *Analyse numérique des équations différentielles*, Masson, Paris, 1989.
- [3] J.D. LAMBERT, *Numerical Methods for Ordinary Differential Systems-The Initial Value Problem*, John Wiley & Sons, 1990.
- [4] F. PĂTRULESCU, *A numerical method for the solution of an autonomous initial value problem*, Carpathian J. Math., **28**, no. 2, pp. 289–296, 2012.
- [5] I. PĂVĂLOIU, *On an approximation formula*, Rev. Anal. Numér. Théor. Approx., **26**, no. 1–2, pp. 179–183, 1997. [✉](#)
- [6] A. RALSTON, *Runge-Kutta methods with minimum error bounds*, Math. Comp., **16**, no. 80, pp. 431–437, 1962.
- [7] J. F. TRAUB, *Iterative Methods for the Solution of Equations*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1964.

Received by the editors: June 12, 2012.