



Interval methods of Adams-Bashforth type with variable step sizes

Andrzej Marciniak^{1,2} · Malgorzata A. Jankowska³ 

Received: 22 March 2019 / Accepted: 2 July 2019 / Published online: 29 July 2019
© The Author(s) 2019

Abstract

In a number of our previous papers, we have proposed interval versions of multistep methods (explicit and implicit), including interval predictor-corrector methods, in which the step size was constant. In this paper, we present interval versions of Adams-Bashforth methods with a possibility to change step sizes. This possibility can be used to obtain interval enclosures of the exact solution with a width given beforehand.

Keywords Initial value problem · Adams-Bashforth methods · Interval Adams-Bashforth methods · Variable step size · Floating-point interval arithmetic

1 Introduction

Using computers and approximate methods to solve problems described in the form of ordinary differential equations, we usually provide all calculations in floating-point arithmetic. This arithmetic causes two kinds of errors: representation errors (most real numbers cannot be represented exactly) and rounding errors (differences between the calculated approximations of numbers and their exact mathematical values), which occur during floating-point operations. Using approximate methods, we introduce the third kind of errors—the errors of methods, usually called the truncation errors.

✉ Malgorzata A. Jankowska
malgorzata.jankowska@put.poznan.pl

Andrzej Marciniak
andrzej.marciniak@put.poznan.pl

- ¹ Institute of Computing Science, Poznan University of Technology, Piotrowo 2, 60-965, Poznan, Poland
- ² Department of Computer Science, State University of Applied Sciences in Kalisz, Poznanska 201-205, 62-800, Kalisz, Poland
- ³ Institute of Applied Mechanics, Poznan University of Technology, Jana Pawła II 24, 60-965, Poznan, Poland

Interval arithmetic (see, e.g., [12, 31, 32, 37]) realized in floating-point computer arithmetic is a way to estimate two first kinds of errors. Applying interval methods to approximate the solution of the initial value problem in floating-point interval arithmetic (see, e.g., [11]) we can obtain enclosures of the solution in the form of intervals which contain both these errors and also the truncation error.

We can distinguish three main kinds of interval methods for solving the initial value problem: methods based on high-order Taylor series (see, e.g., [1, 2, 6, 15, 33, 34]), explicit and implicit methods of Runge-Kutta type ([9, 10, 22, 28, 30, 37]), and explicit and implicit multistep methods ([9, 17–19, 22, 25, 26]), including interval predictor-corrector methods [27]. In interval methods based on Runge-Kutta and in interval multistep methods considered so far, a constant step size has been used. The methods based on high-order Taylor series use variable step sizes and seem to be the most universal. But it should be noted that in [25–27] and [30] we have shown that in some cases the interval methods of the second and third kinds give better enclosures of the exact solutions. Therefore, it is worth to take into account also such methods.

In conventional multistep methods, a problem for changing step size was described firstly by F. Ceschino in [4], and developed further, among others, by C. V. D. Forrington [8] and F. T. Krogh [20]. In these methods by appropriate selection of step size, one can minimize the truncation error, i.e., obtain an approximate solution at some points (mesh points of a grid) with a tolerance given beforehand. Since in interval methods the truncation errors are included in interval enclosures of the exact solutions, the only reason to change a given step size is to decrease the widths of these enclosures. Although we consider only interval methods of Adams-Bashforth type, it seems that the presented procedure can be also applied to other interval multistep methods.

The paper is divided into seven sections. In Section 2, we shortly recall conventional Adams-Bashforth methods with variable step sizes. Interval versions of these methods are proposed in Section 3. Section 4 is the main section of this paper, in which we describe how to match the step size to obtain interval enclosures of the exact solution with a desired width. Since for simplicity in Sections 2–4 we consider only one-dimensional initial value problem, in Section 5, we announce some remarks on solving systems of differential equations by interval Adams-Bashforth methods using our algorithm. In Section 6, we present some numerical examples, and in the last section, some conclusions are given.

2 Adams-Bashforth methods

Below we present briefly (on the basis of [11, p. III.5]) the methods of Adams-Bashforth (explicit multistep methods)¹ with variable step sizes.

¹The term “Adams-Bashforth method” is often used only for the four-step explicit linear method, which is a special case of the class of methods that we consider in this paper. In [11], the methods considered here are called “explicit Adams methods,” but J. C. Butcher in [3, p. 143] used the same term as we use, since there are other similar linear explicit methods known as Adams-Nyström or simply Nyström methods (see, e.g., [3] or [13]).

Let us consider the initial value problem²

$$y' = f(t, y), \quad y(0) = y_0, \quad t \in [0, a]. \quad (1)$$

Let the set of points $\{t_0, t_1, \dots, t_k, \dots, t_m\}$, such that

$$0 = t_0 < t_1 < \dots < t_k < \dots < t_m \leq a,$$

be a grid on the interval $[0, a]$ (the points t_i are called the mesh points), and $h_i = t_i - t_{i-1}$ ($i = 1, 2, \dots, m$) denote the step sizes. On the interval $[t_{k-1}, t_k]$, the differential equation (1) is equivalent to

$$y(t_k) = y(t_{k-1}) + \int_{t_{k-1}}^{t_k} f(\tau, y(\tau)) d\tau. \quad (2)$$

The integrand in (2) we approximate by the interpolation polynomial $P(\tau)$ of degree $n - 1$ in such a way that

$$P(t_{k-j}) = f(t_{k-j}, y(t_{k-j})), \quad j = 1, 2, \dots, n.$$

Substituting $\tau = t_{k-1} + th_k$ and using Newton's interpolation formula, we can write the polynomial $P(\tau)$ in the form

$$\begin{aligned} P(t_{k-1} + th_k) &= f(t_{k-1}, y(t_{k-1})) + (t - t_{k-1}) [t_{k-1}, t_{k-2}; f] \\ &\quad + (t - t_{k-1})(t - t_{k-2}) [t_{k-1}, t_{k-2}, t_{k-3}; f] \\ &\quad + \dots + (t - t_{k-1})(t - t_{k-2}) \dots (t - t_{k-n+1}) [t_{k-1}, t_{k-2}, \dots, t_{k-n}; f] \\ &= \sum_{j=0}^{n-1} [t_{k-1}, t_{k-2}, \dots, t_{k-j-1}; f] \prod_{i=0}^{j-1} (t - t_{k-i-1}), \end{aligned}$$

where $[t_{k-1}, t_{k-2}, \dots, t_{k-j-1}; f]$ denote the divided differences given by

$$\begin{aligned} [t_{k-1}, t_{k-2}, \dots, t_{k-j-1}; f] &= \sum_{i=k-1}^{k-j-1} \frac{f(t_i, y(t_i))}{\prod_{\substack{l=k-1 \\ l \neq i}}^{k-j-1} (t_i - t_l)}, \\ j &= 1, 2, \dots, n-1, \end{aligned} \quad (3)$$

which may be defined recursively by

$$\begin{aligned} [t_{k-1}; f] &= f(t_{k-1}, y(t_{k-1})), \\ [t_{k-1}, t_{k-2}, \dots, t_{k-j-1}; f] &= \frac{[t_{k-1}, t_{k-2}, \dots, t_{k-j}; f] - [t_{k-2}, t_{k-3}, \dots, t_{k-j-1}; f]}{t_{k-1} - t_{k-j-1}}, \\ j &= 1, 2, \dots, n-1. \end{aligned}$$

²The case when (1) presents a system of differential equations is discussed in Section 5

Let us denote

$$\begin{aligned}\varphi_0(k) &= f(t_{k-1}, y(t_{k-1})), \\ \varphi_j(k) &= [t_{k-1}, t_{k-2}, \dots, t_{k-j-1}; f] \prod_{i=0}^{j-1} (t_k - t_{k-i-1}),\end{aligned}\quad (4)$$

$$\begin{aligned}g_0(k) &= 1, \\ g_j(k) &= \frac{1}{h_k} \int_{t_{k-1}}^{t_k} \prod_{i=0}^{j-1} \frac{t - t_{k-i-1}}{t_k - t_{k-i-1}} dt.\end{aligned}\quad (5)$$

Then, the formula (2) may be written as

$$y(t_k) = y(t_{k-1}) + h_k \sum_{j=0}^{n-1} g_j(k) \varphi_j(k) + \int_{t_{k-1}}^{t_k} r_n(\tau) d\tau, \quad (6)$$

where $r_n(\tau)$ denotes the interpolation error, and the integral over this error is the local truncation error. We have (see, e.g., [14, 35])

$$\begin{aligned}\int_{t_{k-1}}^{t_k} r_n(\tau) d\tau &= \frac{y^{(n+1)}(\xi)}{n!} \int_{t_{k-1}}^{t_k} \prod_{i=1}^n (\tau - t_{k-i}) d\tau \\ &= \frac{y^{(n+1)}(\xi)}{n!} h_k^{n+1} \int_0^1 t \prod_{i=2}^n \left(t + \frac{t_{k-1} - t_{k-i}}{h_k}\right) dt,\end{aligned}$$

where $\xi \in [t_{k-n}, t_k]$ and $y^{(n+1)}(\xi) = f^{(n)}(\xi, y(\xi))$. Thus, the formula (6) may be written in the form

$$y(t_k) = y(t_{k-1}) + h_k \sum_{j=0}^{n-1} g_j(k) \varphi_j(k) + h_k^{n+1} g_n(k) f^{(n)}(\xi, y(\xi)), \quad (7)$$

where the values $\varphi_j(k)$ and $g_j(k)$ for $j = 0, 1, \dots, n-1$ are given by (4) and (5), respectively, and

$$g_n(k) = \frac{1}{n!} \int_0^1 t \prod_{i=2}^n \left(t + \frac{t_{k-1} - t_{k-i}}{h_k}\right) dt. \quad (8)$$

The values $\varphi_j(k)$ and $g_j(k)$ for $j = 0, 1, \dots, n-1$ can be computed efficiently with recurrence relations [11, 13]. As an immediate consequence of (3), we have

$$\begin{aligned}\varphi_0(k) &= \bar{\varphi}_0(k) = f(t_{k-1}, y(t_{k-1})), \\ \bar{\varphi}_j(k) &= \bar{\varphi}_{j-1}(k) - \varphi_{j-1}(k-1), \\ \varphi_j(k) &= \beta_j(k) \bar{\varphi}_j(k), \quad j = 1, 2, \dots, n-1,\end{aligned}$$

where the coefficients

$$\beta_j(k) = \prod_{i=0}^{j-1} \frac{t_k - t_{k-i-1}}{t_{k-1} - t_{k-i-2}}$$

can be calculated by

$$\begin{aligned} \beta_0(k) &= 1, \\ \beta_j(k) &= \beta_{j-1}(k) \frac{t_k - t_{k-j}}{t_{k-1} - t_{k-j-1}}, \quad j = 1, 2, \dots, n-1. \end{aligned}$$

The calculation of the coefficients $g_j(k)$ is more tricky [13, 21]. We introduce the q -fold integral

$$c_{jq}(t) = \frac{(q-1)!}{h_k^q} \int_{t_{k-1}}^t \int_{t_{k-1}}^{\xi_{q-1}} \dots \int_{t_{k-1}}^{\xi_1} \prod_{i=0}^{j-1} \frac{\xi_0 - t_{k-i-1}}{t_k - t_{k-i-1}} d\xi_0 d\xi_1 \dots d\xi_{q-1}.$$

It appears that

$$g_j(k) = c_{j1}(t_k), \quad j = 1, 2, \dots, n-1,$$

where for the quantities $c_{jq}(t_k)$ we have the following recurrence relations:

$$\begin{aligned} c_{0q}(t_k) &= \frac{1}{q}, \quad c_{1q}(t_k) = \frac{1}{q(q+1)}, \\ c_{jq}(t_k) &= c_{j-1,q}(t_k) - c_{j-1,q+1}(t_k) \frac{h_k}{t_k - t_{k-j}}. \end{aligned}$$

If the approximations

$$y_{k-1}, y_{k-2}, \dots, y_{k-n}$$

of the exact values

$$y(t_{k-1}), y(t_{k-2}), \dots, y(t_{k-n})$$

are known, we denote $f_i = f(t_i, y_i)$ for $i = k-n, k-n+1, \dots, k-1$, and in the formula (7) we omit the truncation error, then we obtain the Adams-Bashforth methods with variable step sizes given by

$$y_k = y_{k-1} + h_k \sum_{j=0}^{n-1} g_j(k) \varphi_j(k). \quad (9)$$

In particular, from (9) we get:

$$- \quad n = 1$$

$$y_k = y_{k-1} + h_k f_{k-1}, \quad (10)$$

$$- \quad n = 2$$

$$y_k = y_{k-1} + h_k \gamma_{2k}, \quad (11)$$

where

$$\gamma_{2k} = f_{k-1} + \frac{1}{2} \frac{h_k}{h_{k-1}} (f_{k-1} - f_{k-2}),$$

$$- \quad n = 3$$

$$y_k = y_{k-1} + h_k \gamma_{3k}, \quad (12)$$

where

$$\begin{aligned} \gamma_{3k} = \gamma_{2k} + \frac{1}{2} \left(1 - \frac{1}{3} \frac{h_k}{h_k + h_{k-1}} \right) \frac{h_k}{h_{k-1}} \frac{h_k + h_{k-1}}{h_{k-1} + h_{k-2}} \\ \times \left[f_{k-1} - f_{k-2} - \frac{h_{k-1}}{h_{k-2}} (f_{k-2} - f_{k-3}) \right], \end{aligned}$$

$$- \quad n = 4$$

$$y_k = y_{k-1} + h_k \gamma_{4k}, \quad (13)$$

where

$$\begin{aligned} \gamma_{4k} = \gamma_{3k} \\ + \left[\frac{1}{2} \left(1 - \frac{1}{3} \frac{h_k}{h_k + h_{k-1}} \right) - \frac{1}{6} \left(1 - \frac{1}{2} \frac{h_k}{h_k + h_{k-1}} \right) \frac{h_k}{h_k + h_{k-1} + h_{k-2}} \right] \\ \times \frac{h_k}{h_{k-1}} \frac{h_k + h_{k-1}}{h_{k-1} + h_{k-2}} \frac{h_k + h_{k-1} + h_{k-2}}{h_{k-1} + h_{k-2} + h_{k-3}} \\ \times \left\{ f_{k-1} - f_{k-2} - \frac{h_{k-1}}{h_{k-2}} (f_{k-2} - f_{k-3}) \right. \\ \left. - \frac{h_{k-1}}{h_{k-2}} \frac{h_{k-1} + h_{k-2}}{h_{k-2} + h_{k-3}} \left[f_{k-2} - f_{k-3} - \frac{h_{k-2}}{h_{k-3}} (f_{k-3} - f_{k-4}) \right] \right\}. \end{aligned}$$

If for each $i = k, k-1, \dots, k-n+1$ we have $h_i = h$ (a constant step size), then from the above formulas we obtain the following well-known conventional Adams-Bashforth methods (see, e.g., [3, 5, 11, 16, 38]):

$$- \quad n = 1 \text{ (Euler's method)}$$

$$y_k = y_{k-1} + h f_{k-1},$$

$$- \quad n = 2$$

$$y_k = y_{k-1} + \frac{h}{2} (3f_{k-1} - f_{k-2}),$$

$$- \quad n = 3$$

$$y_k = y_{k-1} + \frac{h}{12} (23f_{k-1} - 16f_{k-2} + 5f_{k-3}),$$

$$- \quad n = 4$$

$$y_k = y_{k-1} + \frac{h}{24} (55f_{k-1} - 59f_{k-2} + 37f_{k-3} - 9f_{k-4}).$$

3 Interval versions of Adams-Bashforth methods with variable step sizes

Let us denote:

$$- \quad \Delta_t \text{ and } \Delta_y \text{—bounded sets in which the function } f(t, y), \text{ occurring in (1), is defined, i.e.,}$$

$$\Delta_t = \{t \in \mathbb{R} : 0 \leq t \leq a\}, \quad \Delta_y = \{y \in \mathbb{R} : \underline{b} \leq y \leq \bar{b}\},$$

- $F(T, Y)$ —an interval extension of $f(t, y)$, where an interval extension of the function

$$f : \mathbb{R} \times \mathbb{R} \supset \Delta_t \times \Delta_y \rightarrow \mathbb{R}$$

we call a function

$$F : \mathbb{IR} \times \mathbb{IR} \supset \mathbb{I}\Delta_t \times \mathbb{I}\Delta_y \rightarrow \mathbb{IR}$$

such that

$$(t, y) \in (T, Y) \Rightarrow f(t, y) \in F(T, Y),$$

where \mathbb{IR} denotes the space of real intervals.

- $\Psi(T, Y)$ —an interval extension of $f^{(n)}(t, y(t)) \equiv y^{(n+1)}(t)$,

and let us assume that:

- the function $F(T, Y)$ is defined and continuous³ for all $T \subset \Delta_t$ and $Y \subset \Delta_y$,
- the function $F(T, Y)$ is monotonic with respect to inclusion, i.e.,

$$T_1 \subset T_2 \wedge Y_1 \subset Y_2 \Rightarrow F(T_1, Y_1) \subset F(T_2, Y_2),$$

- for each $T \subset \Delta_t$ and for each $Y \subset \Delta_y$, there exists a constant $\Lambda > 0$ such that

$$w(F(T, Y)) \leq \Lambda (w(T) + w(Y)),$$

where $w(A)$ denotes the width of the interval A ,

- the function $\Psi(T, Y)$ is defined for all $T \subset \Delta_t$ and $Y \subset \Delta_y$,
- the function $\Psi(T, Y)$ is monotonic with respect to inclusion.

Moreover, let us assume that $y(0) \in Y_0$ and the intervals Y_k such that $y(t_k) \in Y_k$ for $k = 1, 2, \dots, n-1$ are known. We can obtain such Y_k by applying an interval one-step method, for example, an interval method of Runge-Kutta type (see, e.g., [9, 10, 22, 28, 30, 37]).

The interval methods of Adams-Bashforth type with variable step sizes we define as follows (compare (7)):

$$\begin{aligned} Y_k = & Y_{k-1} + h_k \sum_{j=0}^{n-1} g_j(k) \Phi_j(k) \\ & + h_k^{n+1} g_n(k) \Psi(T_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k], \\ & Y_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k] \\ & \times F(\Delta_t, \Delta_y)), \\ & k = n, n+1, \dots, m, \end{aligned} \tag{14}$$

³The function $F(T, Y)$ is continuous at (T_0, Y_0) if for every $\varepsilon > 0$ there is a positive number $\delta = \delta(\varepsilon)$ such that $d(F(T, Y), F(T_0, Y_0)) < \varepsilon$ whenever $d(T, T_0) < \delta$ and $d(Y, Y_0) < \delta$. Here, d denotes the interval metric defined by $d(X_1, X_2) = \max\{|X_1 - X_2|, |\bar{X}_1 - \bar{X}_2|\}$, where $X_1 = [\underline{X}_1, \bar{X}_1]$ and $X_2 = [\underline{X}_2, \bar{X}_2]$ are two intervals.

where $g_j(k)$ for $j = 1, 2, \dots, n-1$ is given by (5), $g_n(k)$ —by (8),

$$\begin{aligned}\Phi_0(k) &= F_{k-1}, \\ \Phi_j(k) &= [t_{k-1}, t_{k-2}, \dots, t_{k-j-1}; F] \prod_{i=0}^{j-1} (t_k - t_{k-i-1}), \\ [t_{k-1}, t_{k-2}, \dots, t_{k-j-1}; F] &= \sum_{i=k-1}^{k-j-1} \frac{F_i}{\prod_{\substack{l=k-1 \\ l \neq i}} (t_i - t_l)}, \\ j &= 1, 2, \dots, n-1,\end{aligned}\quad (15)$$

and $F_i = F(T_i, Y_i)$ for $i = k-n, k-n+1, \dots, k-1$.

In particular, for a given n , we get the following methods (compare (10)–(13)):

– $n = 1$

$$\begin{aligned}Y_k &= Y_{k-1} + h_k F_{k-1} \\ &\quad + \frac{h_k^2}{2} \Psi(T_{k-1} + [0, h_k], Y_{k-1} + [0, h_k] F(\Delta_t, \Delta_y)),\end{aligned}\quad (16)$$

– $n = 2$

$$\begin{aligned}Y_k &= Y_{k-1} + h_k \Gamma_{2k} \\ &\quad + h_k^3 g_2(k) \Psi(T_{k-1} + [-h_{k-1}, h_k], \\ &\quad Y_{k-1} + [-h_{k-1}, h_k] F(\Delta_t, \Delta_y)),\end{aligned}\quad (17)$$

where

$$\begin{aligned}\Gamma_{2k} &= F_{k-1} + \frac{1}{2} \frac{h_k}{h_{k-1}} (F_{k-1} - F_{k-2}), \\ g_2(k) &= \frac{1}{2} \left(\frac{1}{3} + \frac{1}{2} \frac{h_{k-1}}{h_k} \right),\end{aligned}$$

– $n = 3$

$$\begin{aligned}Y_k &= Y_{k-1} + h_k \Gamma_{3k} \\ &\quad + h_k^4 g_3(k) \Psi(T_{k-1} + [-h_{k-2} - h_{k-1}, h_k], \\ &\quad Y_{k-1} + [-h_{k-2} - h_{k-1}, h_k] F(\Delta_t, \Delta_y)),\end{aligned}\quad (18)$$

where

$$\begin{aligned}\Gamma_{3k} &= \Gamma_{2k} \\ &\quad + \frac{1}{2} \left(1 - \frac{1}{3} \frac{h_k}{h_k + h_{k-1}} \right) \frac{h_k}{h_{k-1}} \frac{h_k + h_{k-1}}{h_{k-1} + h_{k-2}} \\ &\quad \times \left[F_{k-1} - F_{k-2} - \frac{h_{k-1}}{h_{k-2}} (F_{k-2} - F_{k-3}) \right], \\ g_3(k) &= \frac{1}{6} \left(\frac{1}{4} + \frac{1}{3} \frac{2h_{k-1} + h_{k-2}}{h_k} + \frac{1}{2} \frac{h_{k-1}}{h_k} \frac{h_{k-1} + h_{k-2}}{h_k} \right),\end{aligned}$$

– $n = 4$

$$\begin{aligned} Y_k &= Y_{k-1} + h_k \Gamma_{4k} \\ &\quad + h_k^5 g_4(k) \Psi \left(T_{k-1} + [-h_{k-3} - h_{k-2} - h_{k-1}, h_k], \right. \\ &\quad \left. Y_{k-1} + [-h_{k-3} - h_{k-2} - h_{k-1}, h_k] F(\Delta_t, \Delta_y) \right), \end{aligned} \quad (19)$$

where

$$\begin{aligned} \Gamma_{4k} &= \Gamma_{3k} \\ &\quad + \left[\frac{1}{2} \left(1 - \frac{1}{3} \frac{h_k}{h_k + h_{k-1}} \right) - \frac{1}{6} \left(1 - \frac{1}{2} \frac{h_k}{h_k + h_{k-1}} \right) \frac{h_k}{h_k + h_{k-1} + h_{k-2}} \right] \\ &\quad \times \frac{h_k}{h_{k-1}} \frac{h_k + h_{k-1}}{h_{k-1} + h_{k-2}} \frac{h_k + h_{k-1} + h_{k-2}}{h_{k-1} + h_{k-2} + h_{k-3}} \\ &\quad \times \left\{ F_{k-1} - F_{k-2} - \frac{h_{k-1}}{h_{k-2}} (F_{k-2} - F_{k-3}) \right. \\ &\quad \left. - \frac{h_{k-1}}{h_{k-2}} \frac{h_{k-1} + h_{k-2}}{h_{k-2} + h_{k-3}} \left[F_{k-2} - F_{k-3} - \frac{h_{k-2}}{h_{k-3}} (F_{k-3} - F_{k-4}) \right] \right\}, \\ g_4(k) &= \frac{1}{24} \left(\frac{1}{5} + \frac{1}{4} \frac{3h_{k-1} + 2h_{k-2} + h_{k-3}}{h_k} \right. \\ &\quad + \frac{1}{3} \frac{h_{k-1}}{h_k} \frac{3h_{k-1} + 4h_{k-2} + 2h_{k-3}}{h_k} + \frac{1}{3} \frac{h_{k-2}}{h_k} \frac{h_{k-2} + h_{k-3}}{h_k} \\ &\quad \left. + \frac{1}{2} \frac{h_{k-1}}{h_k} \frac{h_{k-1} + h_{k-2}}{h_k} \frac{h_{k-1} + h_{k-2} + h_{k-3}}{h_k} \right). \end{aligned}$$

For the methods (14), we can prove that the exact solution of the initial value problem (1) belongs to the intervals (enclosures) obtained by these methods. Before that, it is convenient to present the following

Lemma *If $(t_i, y(t_i)) \in (T_i, Y_i)$ for $i = k-n, k-n+1, \dots, k-1$, where $Y_i = Y(t_i)$, then for any $j = 0, 1, \dots, n-1$ we have*

$$\varphi_j(k) \in \Phi_j(k). \quad (20)$$

Proof Since $F(T, Y)$ is an interval extension of $f(t, y)$ and $(t_i, y(t_i)) \in (T_i, Y_i)$ for $i = k-n, k-n+1, \dots, k-1$, we can write

$$f(t_{k-1-l}, y(t_{k-1-l})) \in F(T_{k-1-l}, Y_{k-1-l}), \quad l = 0, 1, \dots, j.$$

This implies that

$$\sum_{i=k-1}^{k-j-1} \frac{f(t_i, y(t_i))}{\prod_{\substack{l=k-1 \\ l \neq i}}^{k-j-1} (t_i - t_l)} \in \sum_{i=k-1}^{k-j-1} \frac{F_i}{\prod_{\substack{l=k-1 \\ l \neq i}}^{k-j-1} (t_i - t_l)}. \quad (21)$$

From (15) and (21), the relation (20) follows immediately. \square

Theorem 1 If $y(0) \in Y_0$ and $y(t_i) \in Y_i$ for $i = 1, 2, \dots, n-1$, then for the exact solution $y(t)$ of the initial value problem (1) we have

$$y(t_k) \in Y_k \quad (22)$$

for $k = n, n+1, \dots, m$, where $Y_k = Y(t_k)$ are obtained from the methods (14).

Proof Let us consider the formula (7) for $k = n$. We have

$$y(t_n) = y(t_{n-1}) + h_n \sum_{j=0}^{n-1} g_j(n) \varphi_j(n) + h_n^{n+1} g_n(n) f^{(n)}(\xi, y(\xi)), \quad (23)$$

where $\xi \in [t_0, t_n]$. From the assumption we get $y(t_{n-1}) \in Y_{n-1}$, and from the Lemma it follows that

$$h_n \sum_{j=0}^{n-1} g_j(n) \varphi_j(n) \in h_n \sum_{j=0}^{n-1} g_j(n) \Phi_j(n).$$

Applying Taylor's formula, we get

$$y(\xi) = y(t_{n-1}) + (\xi - t_{n-1}) y'(t_{n-1} + \vartheta(\xi - t_{n-1})), \quad (24)$$

where $\vartheta \in [0, 1]$. Due to $\xi \in [t_0, t_n]$ and $t_i = h_1 + h_2 + \dots + h_i$ for $i = 1, 2, \dots, m$ ($t_0 = 0$), we have

$$\xi - t_{n-1} \in [-h_1 - h_2 - \dots - h_{n-1}, h_n]. \quad (25)$$

Moreover, $y'(t) = f(t, y(t))$. Since

$$f[t_{n-1} + \vartheta(\xi - t_{n-1}), y(t_{n-1} + \vartheta(\xi - t_{n-1}))] \in F(\Delta_t, \Delta_y),$$

then

$$y'(t_{n-1} + \vartheta(\xi - t_{n-1})) \in F(\Delta_t, \Delta_y).$$

Taking into account the above considerations, from (24), we get

$$y(\xi) \in Y_{n-1} + [-h_1 - h_2 - \dots - h_{n-1}, h_n] F(\Delta_t, \Delta_y). \quad (26)$$

As we assumed, Ψ is an interval extension of $f^{(n)}(t, y)$. Hence, applying (25) and (26), we have

$$\begin{aligned} & h_n^{n+1} g_n(n) f^{(n)}(\xi, y(\xi)) \\ & \in h_n^{n+1} g_n(n) \Psi(T_{n-1} + [-h_1 - h_2 - \dots - h_{n-1}, h_n], \\ & \quad Y_{n-1} + [-h_1 - h_2 - \dots - h_{n-1}, h_n]) F(\Delta_t, \Delta_y). \end{aligned}$$

Thus, we have shown that $y(t_n)$ belongs to the interval

$$\begin{aligned} & Y_{n-1} + h_n \sum_{j=0}^{n-1} g_j(n) \Phi_j(n) \\ & + h_n^{n+1} g_n(n) \Psi(T_{n-1} + [-h_1 - h_2 - \dots - h_{n-1}, h_n], \\ & \quad Y_{n-1} + [-h_1 - h_2 - \dots - h_{n-1}, h_n]) F(\Delta_t, \Delta_y), \end{aligned}$$

but—according to (14)—this is the interval Y_n . This conclusion ends the proof for $k = n$. In a similar way we can show the thesis of this theorem for $k = n+1, n+2, \dots, m$. \square

Theorem 2 *If the intervals Y_k for $k = 0, 1, \dots, n-1$ are known, $t_i = h_1 + h_2 + \dots + h_i \in T_i$, $i = 1, 2, \dots, m$ ($t_0 = 0 \in T_0 = [0, 0]$), and the intervals Y_k for $k = n, n+1, \dots, m$ are obtained from (14), then*

$$w(Y_k) \leq A \max_{q=0,1,\dots,n-1} w(Y_q) + B \max_{j=1,2,\dots,m-1} w(T_j) + Ch^n, \quad (27)$$

where

$$h = \max_{k=n,n+1,\dots,m} h_k, \quad (28)$$

and the constants A , B , and C are independent of h .

Proof Because of $g_j(k) > 0$ for $j = 0, 1, \dots, n-1$ (see (5)) and $g_n(k) > 0$ (see (8)), from (14), we get

$$\begin{aligned} w(Y_k) &\leq w(Y_{k-1}) + h_k \sum_{j=0}^{n-1} g_j(k) w(\Phi_j(k)) \\ &\quad + h_k^{n+1} g_n(k) \\ &\quad \times w(\Psi(T_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k], \\ &\quad Y_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k]) \\ &\quad \times F(\Delta_t, \Delta_y)). \end{aligned} \quad (29)$$

We assumed that Ψ is monotonic with respect to inclusion. Moreover, if step sizes h_{k-j} for $j = 0, 1, \dots, n-1$ are such that satisfy the conditions

$$\begin{aligned} T_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k] &\subset \Delta_t, \\ Y_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k] &\subset \Delta_y, \end{aligned}$$

then

$$\begin{aligned} \Psi(T_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k], \\ Y_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k]) &\subset \Psi(\Delta_t, \Delta_y). \end{aligned}$$

From the above inclusion, it follows that

$$\begin{aligned} w(\Psi(T_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k], \\ Y_{k-1} + [-h_{k-n+1} - h_{k-n+2} - \dots - h_{k-1}, h_k]) \\ \times F(\Delta_t, \Delta_y)) \leq w(\Psi(\Delta_t, \Delta_y)). \end{aligned} \quad (30)$$

If we denote

$$\begin{aligned} \alpha_{i0}(k) = 1, \quad \alpha_{ij}(k) &= \frac{\prod_{l=0}^{j-1} (t_k - t_{k-l-1})}{\prod_{\substack{l=k-1 \\ l \neq i}}^{k-j-1} (t_i - t_l)}, \\ i &= k-1, k-2, \dots, k-j-1, \quad j = 1, 2, \dots, n-1, \end{aligned}$$

then from (15) we have

$$\Phi_j(k) = \sum_{i=k-1}^{k-j-1} \alpha_{ij}(k) F_i. \quad (31)$$

But we also assumed that for the function F there exists a constant $\Lambda > 0$ such that

$$w(F_i) \leq \Lambda (w(T_i) + w(Y_i)). \quad (32)$$

Therefore, from (29)–(32), we get

$$\begin{aligned} w(Y_k) &\leq w(Y_{k-1}) + h_k \Lambda \sum_{j=0}^{n-1} g_j(k) \sum_{i=k-1}^{k-j-1} |\alpha_{ij}(k)| (w(T_i) + w(Y_i)) \\ &\quad + h_k^{n+1} g_n(k) w(\Psi(\Delta_t, \Delta_y)). \end{aligned}$$

If we denote

$$\alpha_j(k) = \max_{i=k-1, k-2, \dots, k-j-1} |\alpha_{ij}(k)|,$$

then from the above inequality it follows that

$$\begin{aligned} w(Y_k) &\leq w(Y_{k-1}) + h_k \Lambda \sum_{j=0}^{n-1} g_j(k) \alpha_j(k) \sum_{i=k-1}^{k-j-1} (w(T_i) + w(Y_i)) \\ &\quad + h_k^{n+1} g_n(k) w(\Psi(\Delta_t, \Delta_y)), \end{aligned}$$

and denoting

$$\rho_n(k) = \max_{j=0, 1, \dots, n-1} \alpha_j(k), \quad (33)$$

we have

$$\begin{aligned} w(Y_k) &\leq w(Y_{k-1}) + h_k \Lambda \rho_n(k) \sum_{j=0}^{n-1} \sum_{i=k-1}^{k-j-1} (w(T_i) + w(Y_i)) \\ &\quad + h_k^{n+1} g_n(k) w(\Psi(\Delta_t, \Delta_y)). \end{aligned} \quad (34)$$

But

$$\sum_{j=0}^{n-1} \sum_{i=k-1}^{k-j-1} (w(T_i) + w(Y_i)) = \sum_{j=1}^n (n-j+1) (w(T_{k-j}) + w(Y_{k-j})).$$

Thus, from (34), we get

$$\begin{aligned} w(Y_k) &\leq w(Y_{k-1}) + \varsigma_n(h_k) \sum_{j=1}^n (n-j+1) w(Y_{k-j}) \\ &\quad + \varsigma_n(h_k) \sum_{j=1}^n (n-j+1) w(T_{k-j}) + \sigma_n(h_k) w(\Psi(\Delta_t, \Delta_y)), \end{aligned} \quad (35)$$

where

$$\varsigma_n(h_k) = h_k \Lambda \rho_n(k), \quad \sigma_n(h_k) = h_k^{n+1} g_n(k). \quad (36)$$

From (35) for $k = n$, we obtain

$$\begin{aligned}
 w(Y_n) &\leq [1 + n\varsigma_n(h_n)] w(Y_{n-1}) + \varsigma_n(h_n) \sum_{j=2}^n (n-j+1) w(Y_{n-j}) \\
 &\quad + \varsigma_n(h_n) \sum_{j=1}^n (n-j+1) w(T_{n-j}) + \sigma_n(h_n) w(\Psi(\Delta_t, \Delta_y)) \\
 &\leq [1 + n\varsigma_n(h_n)] w(Y_{n-1}) + n\varsigma_n(h_n) \sum_{j=2}^n w(Y_{n-j}) \\
 &\quad + n\varsigma_n(h_n) \sum_{j=1}^n w(T_{n-j}) + \sigma_n(h_n) w(\Psi(\Delta_t, \Delta_y)) \\
 &\leq [1 + n\varsigma_n(h_n)] \sum_{j=1}^n w(Y_{n-j}) + n\varsigma_n(h_n) \sum_{j=1}^n w(T_{n-j}) \\
 &\quad + \sigma_n(h_n) w(\Psi(\Delta_t, \Delta_y)), \tag{37}
 \end{aligned}$$

and using mathematical induction, one can prove that for each $i = 1, 2, \dots, m-1$ we have

$$w(Y_{n+i}) \leq P_{ni} \sum_{j=1}^n w(Y_{n-j}) + Q_{ni} \sum_{j=1}^{n+i} w(T_{n+i-j}) + R_{ni} w(\Psi(\Delta_t, \Delta_y)), \tag{38}$$

where

$$\begin{aligned}
 P_{ni} &= [1 + n\varsigma_n(h_{n+i})] (1 + P_{n,i-1}), \quad Q_{ni} = [1 + n\varsigma_n(h_{n+i})] Q_{n,i-1} + n\varsigma_n(h_{n+i}), \\
 R_{ni} &= [1 + n\varsigma_n(h_{n+i})] R_{n,i-1} + \sigma_n(h_{n+i}), \\
 P_{n0} &= 1 + n\varsigma_n(h_n), \quad Q_{n0} = n\varsigma_n(h_n), \quad R_{n0} = \sigma_n(h_n).
 \end{aligned}$$

It is self-evident that

$$\varsigma_n(h_k) \leq \varsigma_n(h) = h\Lambda\rho_n, \quad \sigma_n(h_k) \leq \sigma_n(h) = h^{n+1}g_n, \tag{39}$$

where

$$\rho_n = \max_{k=n, n+1, \dots, m} \rho_n(k), \quad g_n = \max_{k=n, n+1, \dots, m} g_n(k),$$

and where h is given by (28). Hence,

$$\begin{aligned}
 P_{ni} &= \sum_{l=0}^i [1 + n\varsigma_n(h)]^{l+1} \leq \sum_{l=0}^i \exp[(l+1)n\varsigma_n(h)] \leq (i+1) \exp[(i+1)n\varsigma_n(h)] \\
 &\leq (m-n+1) \exp[(m-n+1)n\varsigma_n(h)] \leq m \exp[mn\varsigma_n(h)] \\
 &\leq m \exp(m^2 h \Lambda \rho_n) \leq m \exp(m^2 a \Lambda \rho_n),
 \end{aligned}$$

and

$$\begin{aligned} Q_{ni} &= \sum_{l=0}^i [1 + n\zeta_n(h)]^l n\zeta_n(h) = n\zeta_n(h) \sum_{l=0}^i [1 + n\zeta_n(h)]^l = [1 + n\zeta_n(h)]^{i+1} - 1 \\ &\leq \exp[(i+1)n\zeta_n(h)] - 1 \leq \exp[(m-n+1)n\zeta_n(h)] - 1 \leq \exp[mn\zeta_n(h)] - 1 \\ &\leq \exp(m^2 h \Lambda \rho_n) - 1 \leq \exp(m^2 a \Lambda \rho_n) - 1, \end{aligned}$$

$$\begin{aligned} R_{ni} &\leq \sum_{l=0}^i [1 + n\zeta_n(h)]^l n\sigma_n(h) = n\sigma_n(h) \sum_{l=0}^i [1 + n\zeta_n(h)]^l \\ &= \frac{\sigma_n(h)}{\rho_n(h)} \left\{ [1 + n\zeta_n(h)]^{i+1} - 1 \right\} \leq h^n \frac{g_n}{\Lambda \rho_n} \left[\exp(m^2 a \Lambda \zeta_n) - 1 \right]. \end{aligned}$$

It means that from (38), we have

$$\begin{aligned} w(Y_{n+i}) &\leq nm \exp(m^2 a \Lambda \rho_n) \max_{q=0,1,\dots,n-1} w(Y_q) \\ &\quad + (n+i) \left[\exp(m^2 a \Lambda \rho_n) - 1 \right] \max_{j=0,1,\dots,m-1} w(T_j) \\ &\quad + h^n \frac{g_n}{\Lambda \rho_n} \left[\exp(m^2 a \Lambda \rho_n) - 1 \right] w(\Psi(\Delta_t, \Delta_y)). \end{aligned} \quad (40)$$

Since $i \leq m - n$, then from (40), we finally get

$$w(Y_{n+i}) \leq A \max_{q=0,1,\dots,n-1} w(Y_q) + B \max_{j=0,1,\dots,m-1} w(T_j) + Ch^n \quad (41)$$

for each $i = 1, 2, \dots, m - n$, where

$$\begin{aligned} A &= m^2 \exp(m^2 a \Lambda \rho_n), \quad B = m \left[\exp(m^2 a \Lambda \rho_n) - 1 \right], \\ C &= \frac{g_n}{\Lambda \rho_n} \left[\exp(m^2 a \Lambda \rho_n) - 1 \right]. \end{aligned}$$

Since $T_0 = [0, 0]$, i.e., $w(T_0) = 0$, for $i = 1, 2, \dots, m - n$, the inequality (27) follows immediately from (41). For $i = 0$, the relation (41) follows directly from (37). \square

4 Using variable step sizes

In conventional one- and multistep methods, the change of step size is used to obtain approximate solutions with a given tolerance (to decrease the truncation errors). In the case of multistep methods, it is worth to recommend the methods of Adams with variable step size of predictor-corrector type, in which the explicit methods of Adams-Bashforth are predictors and the implicit methods of Adams-Moulton are correctors. A usable algorithm based on these methods has been proposed by F. T. Krogh in [21], and appropriate procedures in Fortran programming language to implement this algorithm on computers have been published in the book [36] written by L. F. Shampine and M. K. Gordon.

In interval methods, all errors (representation, rounding, and truncation errors) are included in obtained enclosures of solutions. Thus, the only aim to choose an adequate step size is to decrease the width of interval enclosures.

To obtain interval enclosures of Y_k with a width eps giving beforehand, one can consider the inequality (27) and compare the right-hand side of it. But this inequality overestimates the width of Y_k , and a better way to do it is to take into consideration the relation (35).

Let us assume that the right-hand side (34) is equal to eps , i.e.,

$$\begin{aligned} w(Y_{k-1}) + \varsigma_n(h_k) \sum_{j=1}^n (n-j+1) w(Y_{k-j}) \\ + \varsigma_n(h_k) \sum_{j=1}^n (n-j+1) w(T_{k-j}) \\ + \sigma_n(h_k) w(\Psi(\Delta_t, \Delta_y)) = eps. \end{aligned} \quad (42)$$

Since T_{k-j} ($j = 1, 2, \dots, n$) is only an interval representation of t_{k-j} , we have $w(T_{k-j}) \approx 0$. If we assume $w(T_{k-j}) = 0$ and take into account (42), then we have

$$\begin{aligned} h_k^{n+1} g_n(k) w(\Psi(\Delta_t, \Delta_y)) \\ + h_k \Delta \rho_n(k) \sum_{j=1}^n (n-j+1) w(Y_{k-j}) + w(Y_{k-1}) - eps = 0. \end{aligned} \quad (43)$$

The equation (43) presents a polynomial (with respect to h) of degree $n+1$. Denoting this polynomial by $p_{n+1}(h_k)$, we look for $h_k > 0$ such that $p_{n+1}(h_k) = 0$. To find such an h_k , one can use any method for finding roots of polynomials. Applying, for example, the Newton iterations we have

$$h_k^{(s+1)} = h_k^{(s)} - \frac{p_{n+1}(h_k^{(s)})}{p'_{n+1}(h_k^{(s)})}, \quad s = 0, 1, \dots, \quad (44)$$

where $h_k^{(0)}$ is given (we can take $h_k^{(0)} = h_{k-1}$). The process (44) is stopped when

$$\left| h_k^{(s+1)} - h_k^{(s)} \right| < \varepsilon,$$

where ε denotes an accuracy given beforehand.

Note that in general, the real $h_k > 0$, satisfying (43), may not exist. Since in (43) all coefficients near by any power of h_k are positive, such a case occurs, for instance, when

$$w(Y_{k-1}) - eps > 0.$$

If this relation holds, we can try to increase eps or simply stop the calculations. Of course, if the widths of successive Y_k increase, then the successive step sizes h_k must decrease (see Example 2 in Section 6). Moreover, if t_m is the last t_k , for which $t_k < a$, and we want to achieve the end a of integration interval, then for the last integration step we should take $h_{m+1} = a - t_m$.

According to the definitions of $g_n(k)$ and $\rho_n(k)$ (see (8) and (33), respectively), it is difficult to write the general form of $p_{n+1}(h_k)$. For $n \leq 4$, we have the following formulas:

$$- \quad n = 1$$

$$p_2(h_k) = \frac{h_k^2}{12} w(\Psi(\Delta_t, \Delta_y)) + h_k \Lambda w(Y_{k-1}) + w(Y_{k-1}) - eps = 0,$$

$$- \quad n = 2$$

$$p_3(h_k) = \frac{h_k^2}{12} (2h_k + 3h_{k-1}) w(\Psi(\Delta_t, \Delta_y)) \\ + \Lambda q_2(h_k) [2w(Y_{k-1}) + w(Y_{k-2})] + w(Y_{k-1}) - eps = 0,$$

where

$$q_2(h_k) = h_k \cdot \max \left\{ 1, \frac{h_k}{h_{k-1}} \right\},$$

$$- \quad n = 3$$

$$p_4(h_k) = \frac{h_k^2}{72} [3h_k^2 + 4h_k(2h_{k-1} + h_{k-2}) \\ + 6h_{k-1}(h_{k-1} + h_{k-2})] w(\Psi(\Delta_t, \Delta_y)) \\ + \Lambda q_3(h_k) [3w(Y_{k-1}) + 2w(Y_{k-2}) + w(Y_{k-3})] \\ + w(Y_{k-1}) - eps = 0,$$

where

$$q_3(h_k) = h_k \cdot \max \left\{ 1, \frac{h_k}{h_{k-1}}, \frac{h_k(h_k + h_{k-1})}{h_{k-1}h_{k-2}} \right\},$$

$$- \quad n = 4$$

$$p_5(h_k) = \frac{h_k^2}{1440} \left\{ 12h_k^3 + 15h_k^2(3h_{k-1} + 2h_{k-2} + h_{k-3}) \right. \\ + 20h_k[h_{k-1}(h_{k-1} + h_{k-2}) \\ + (2h_{k-1} + h_{k-2})(h_{k-1} + h_{k-2} + h_{k-3})] \\ + 30h_{k-1}(h_{k-1} + h_{k-2})(h_{k-1} + h_{k-2} + h_{k-3}) \} \\ \times w(\Psi(\Delta_t, \Delta_y)) \\ + \Lambda q_4(h_k) [4w(Y_{k-1}) + 3w(Y_{k-2}) + 2w(Y_{k-3}) + w(Y_{k-4})] \\ + w(Y_{k-1}) - eps = 0,$$

where

$$q_4(h_k) = h_k \cdot \max \left\{ 1, \frac{h_k}{h_{k-1}}, \frac{h_k(h_k + h_{k-1})}{h_{k-1}h_{k-2}}, \right. \\ \frac{h_k(h_k + h_{k-1})(h_k + h_{k-1} + h_{k-2})}{h_{k-1}h_{k-2}(h_{k-2} + h_{k-3})}, \\ \left. \frac{h_k(h_k + h_{k-1})(h_k + h_{k-1} + h_{k-2})}{(h_{k-1} + h_{k-2})h_{k-2}h_{k-3}} \right\}.$$

Note that in the case of $n > 1$, the form of $p_{n+1}(h_k)$ depends on which $\alpha_j(k)$ ($j = 0, 1, \dots, n-1$) is chosen (see (33)).

5 A note on a system of differential equations

For simplicity, we have considered only one-dimensional initial value problem. The methods (14) and the algorithm presented in Section 4 can be easily extended to systems of ordinary differential equations, i.e., to the initial value problem of the form

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(0) = \mathbf{y}_0, \quad t \in [0, a],$$

where $\mathbf{y} \in \mathbb{R}^N$ and $\mathbf{f} : [0, a] \times \mathbb{R}^N \rightarrow \mathbb{R}^N$. In this case the formula (14) and the relation (22) in Theorem 1 should be written for each component of the interval vector $\mathbf{Y}_k = [Y_{k,(1)}, Y_{k,(2)}, \dots, Y_{k,(N)}]^T$. In the inequality (27) and in the proof of Theorem 2, any number $w(\mathbf{A})$ means $\max_{i=1,2,\dots,N} w(A_i)$ for any vector

$\mathbf{A} = [A_{(1)}, A_{(2)}, \dots, A_{(N)}]^T$. The same concerns the equation (43) for calculating the step size in k th integration step. An example of two-dimensional problem, for which step size changing is applied, we present in the next section (Example 4).

6 Numerical examples

In the examples presented below, we have used our own implementation of floating-point interval arithmetic in Delphi Pascal. This implementation has been written in the form of a unit called *IntervalArithmetic32and64*, which current version is presented in [23]. This unit take advantage of the Delphi Pascal floating-point *Extended* type and makes it possible to represent any input numerical data in the form of a machine interval, perform all calculations in floating-point interval arithmetic, use some standard interval functions and give results in the form of proper intervals (if the ends of an interval are not the same machine numbers, one can see the difference in the output). All programs written in Delphi Pascal for the examples presented one can find in [24]. We have run these programs on Lenovo® Z51 computer with Intel® Core i7 2.4 GHz processor.

Table 1 Starting values for Adams-Bashforth methods and the problem (45)

k	t_k	h_k	y_k	Y_k
0	0.00	–	≈ 1	$[1.0000000000000000\text{E} + 0000, 1.0000000000000000\text{E} + 0000]$
1	0.08	0.08	≈ 1.0408108	$[1.0408107741923882\text{E} + 0000, 1.0408107741923883\text{E} + 0000]$
2	0.15	0.07	≈ 1.0778842	$[1.0778841508846315\text{E} + 0000, 1.0778841508846315\text{E} + 0000]$
3	0.20	0.05	≈ 1.1051709	$[1.1051709180756476\text{E} + 0000, 1.1051709180756477\text{E} + 0000]$

Table 2 Intervals obtained for the problem (45) at $t = 2$ by the methods (16)–(19)

Method	Y_k	Width	CPU time (s)
(16)	[2.7169134040046282E+0000, 2.7235393714205552E+0000]	$\approx 6.63 \cdot 10^{-3}$	0.042
(17)	[2.7179091592957537E+0000, 2.7187125466868537E+0000]	$\approx 8.03 \cdot 10^{-4}$	0.054
(18)	[2.7182298899088899E+0000, 2.7183323624455116E+0000]	$\approx 1.02 \cdot 10^{-4}$	0.055
(19)	[2.7182739085121117E+0000, 2.7182894852166692E+0000]	$\approx 1.56 \cdot 10^{-5}$	0.076

Example 1 At first, let us consider a commonly used test problem

$$y' = 0.5y, \quad y(0) = 1. \quad (45)$$

with the exact solution

$$y = \exp(0.5t). \quad (46)$$

Let us assume

$$\Delta_t = \{t \in \mathbb{R} : 0 \leq t \leq 2\}, \quad \Delta_y = \{y \in \mathbb{R} : 1 \leq y \leq \overline{2.72}\},$$

where \bar{x} denotes the smallest machine number greater or equal to x (similarly, by \underline{x} , we will denote the largest machine number less or equal to x). From (46), we obtain the results presented in Table 1.⁴

Taking $m = 20$ and $h_1 = 0.08$, $h_2 = 0.07$, $h_3 = 0.05$, $h_4 = 0.09$, $h_5 = 0.08$, $h_6 = 0.07$, $h_7 = 0.10$, $h_8 = 0.08$, $h_9 = 0.14$, $h_{10} = 0.09$, $h_{11} = 0.15$, $h_{12} = 0.11$, $h_{13} = 0.07$, $h_{14} = 0.10$, $h_{15} = 0.15$, $h_{16} = 0.12$, $h_{17} = 0.08$, $h_{18} = 0.12$, $h_{19} = 0.15$, and $h_{20} = 0.10$ by the methods (16)–(19) at $t = 2$, we get enclosures of the exact solution presented in Table 2. The methods are very fast and for all of them the time of calculations (CPU time) is less than 1 s. Of course, for each method, the exact solution $y(t_{20}) \approx 2.7182818284590452$ is within the interval obtained. It should be added that if we use the interval methods of Adams-Bashforth type for greater n and very small step sizes we can obtain intervals with greater widths than presented in Table 2 (see [22, Example 4.5 in Sec. 4.7]). This is caused by a great number of calculations in these methods and by a significant increase of rounding errors following from that, which is not compensated for the method orders.

Example 2 Again, let us take into account the problem (45), but now let us apply the procedure for finding step sizes described in Section 4. For the same starting data as in Example 1, requiring 10^{-8} for interval widths and taking $\Lambda = 0.5$, $\varepsilon = 10^{-18}$, $h_1^{(0)} = 0.08$, $h_2^{(0)} = 0.07$, $h_3^{(0)} = 0.05$, and $h_4^{(0)} = 0.09$ as initial approximations of step size for the first integration step for $n = 1, 2, 3$ and 4, respectively (for the next steps we have accepted $h_k^{(0)} = h_{k-1}$), we have obtained (at the first and the last possible to calculate integration steps) the results presented in Table 3. The CPU times have been equal to 36.019 s for the method (16), 1.152 s for (17), 0.344 s

⁴The values of Y_k are presented in the form obtained in our programs.

Table 3 Intervals obtained for the problem (45) at the first and the last calculated integration steps by the methods (16)–(19) with matching step sizes

Method k	t_k	Y_k	Width
(16)	1	≈ 0.0002 [1.0001078385871569E+0000, 1.0001078385888622E+0000]	$\approx 1.71 \cdot 10^{-12}$
	15852	≈ 1.5766 [2.1996396812768524E+0000, 2.1996396912768528E+0000]	$< 1.00 \cdot 10^{-8}$
(17)	2	≈ 0.0815 [1.0415997191016650E+0000, 1.0415997197462058E+0000]	$\approx 6.45 \cdot 10^{-10}$
	278	≈ 0.8333 [1.5168547254474315E+0000, 1.5168547354474316E+0000]	$< 1.00 \cdot 10^{-8}$
(18)	3	≈ 0.1597 [1.0831027071336879E+0000, 1.0831027083961128E+0000]	$\approx 1.26 \cdot 10^{-9}$
	76	≈ 0.6178 [1.3619093684910934E+0000, 1.3619093784910935E+0000]	$< 1.00 \cdot 10^{-8}$
(19)	4	≈ 0.2529 [1.1348046916372604E+0000, 1.1348046936371020E+0000]	$\approx 2.00 \cdot 10^{-9}$
	71	≈ 0.6259 [1.3674527147839997E+0000, 1.3674527247839998E+0000]	$< 1.00 \cdot 10^{-8}$

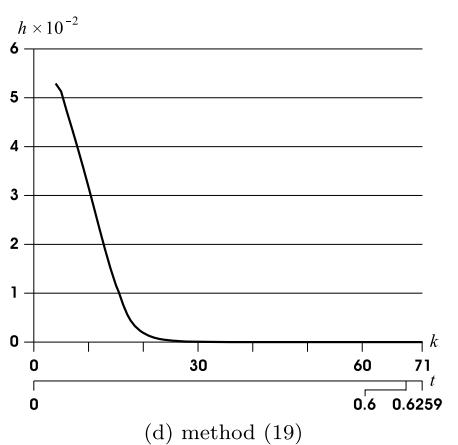
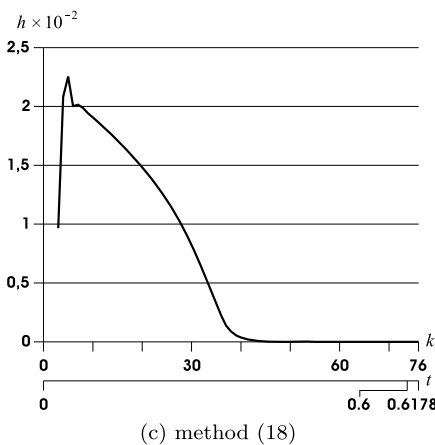
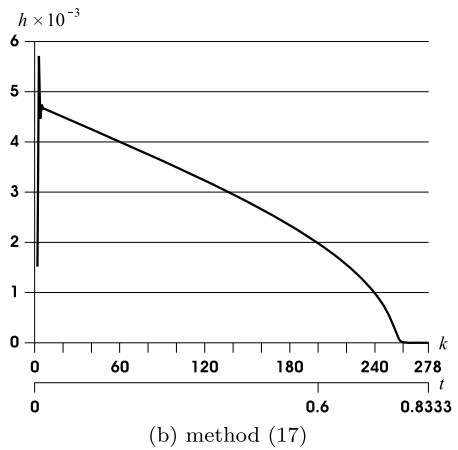
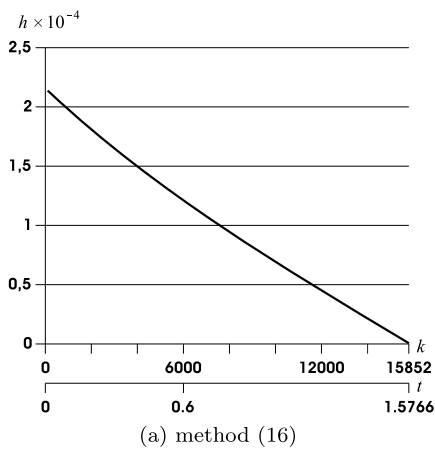


Fig. 1 Step size changes in problem (45) for the methods (16)–(19)

for (18), and 0.415 s for the method (19). The changes of step sizes are shown in Fig. 1a–d.

Let us note that it is not possible to achieve the assumed end $t = 2$ of integration interval. In each method, the step size is decreased for following integration steps (excluding initial steps in two- and more step methods, where some fluctuations can be caused by initial data), but there exists a point t_m , after which further calculations are impossible. To counteract such a situation, one can increase the required widths of intervals or consider a limitation of integration interval. For instance, if for the problem (45) we assume $t \leq 0.6$, then—for the same remaining initial data as previously—we obtain results presented in Table 4 (we present only intervals obtained at the last step for which $t_m < 0.6$ and at the final $t = 0.6$, where the last step size is equal to $0.6 - t_m$). In order to achieved the point $t = 0.6$, the methods (16), (17), (18), and (19) have needed 10.274 s, 0.574 s, 0.138 s, and 0.073 s, respectively. As we could expect, the greater order of method has permitted to obtain results in smaller numbers of integration steps.

Example 3 For the initial value problem of the form (the problem A5 from [7, p. 23])

$$y' = \frac{y-t}{y+t}, \quad y(0) = 4, \quad (47)$$

the solution is unknown. In [7], it is taken $a = 20$. In our methods, we cannot take this value, because for $0 \leq t \leq 20$ we have $-1 < y < 6.3$ and in the interval extension $F(\Delta_t, \Delta_y)$ of $f(x, y)$ for $\Delta_t = \{t \in \mathbf{R} : 0 \leq t \leq 20\}$ and $\Delta_y = \{y \in \mathbf{R} : -1 \leq y \leq 6.3\}$ we have a division by an interval containing zero (such an operation is not defined in proper interval arithmetic). Thus, we restrict these regions to

$$\Delta_t = \{t \in \mathbf{R} : 0 \leq t \leq 10\}, \quad \Delta_y = \{y \in \mathbf{R} : 4 \leq y \leq \overline{6.3}\}.$$

Let us take the fourth-order method (19) with additional starting intervals given in Table 5. These intervals have been obtained by an interval version of conventional

Table 4 Intervals obtained for the problem (45) with $\Delta_t = \{t \in \mathbf{R} : 0 \leq t \leq \overline{0.6}\}$ and $\Delta_y = \{y \in \mathbf{R} : 1 \leq y \leq \overline{2.72}\}$

Method k	t_k	Y_k	Width
(16)	3190 ≈ 0.60000	[1.3498581914855010E+0000, 1.3498581958390336E+0000]	$\approx 4.35 \cdot 10^{-9}$
	3191 0.60000	[1.3498588069670051E+0000, 1.3498588113205398E+0000]	$\approx 4.35 \cdot 10^{-9}$
(17)	135 ≈ 0.59891	[1.3491258950165315E+0000, 1.3491259025297920E+0000]	$\approx 7.51 \cdot 10^{-9}$
	136 0.60000	[1.3498588036394939E+0000, 1.3498588111590695E+0000]	$\approx 7.52 \cdot 10^{-9}$
(18)	32 ≈ 0.59894	[1.3491410173457222E+0000, 1.3491410270971693E+0000]	$\approx 9.75 \cdot 10^{-9}$
	33 0.60000	[1.3498588016695426E+0000, 1.3498588114278859E+0000]	$\approx 9.76 \cdot 10^{-9}$
(19)	15 ≈ 0.59880	[1.3490512149587291E+0000, 1.3490512245357438E+0000]	$\approx 9.58 \cdot 10^{-9}$
	16 0.60000	[1.3498588016932220E+0000, 1.3498588112774857E+0000]	$\approx 9.58 \cdot 10^{-9}$

Table 5 Starting values for the method (19) and the problem (47)

k	t_k	Y_k
0	0	[4.000000000000000E + 0000, 4.000000000000000E + 0000]
1	0.081746227283888863	[4.0801194662264153E + 0000, 4.0801194662264155E + 0000]
2	0.163492454567777736	[4.1571485011046702E + 0000, 4.1571485011046705E + 0000]
3	0.245238681851666599	[4.2313071906453238E + 0000, 4.2313071906453243E + 0000]

Runge-Kutta method (of fourth order) with an optimal step size for the first integration step (see [29] for details). For the method (19), we also need an interval extension of $y^{(5)}(t) = f^{(4)}(t, y)$. Since for the problem (47), we have

$$f^{(4)}(t, y) = \frac{40(y^2 + t^2)}{(y + t)^9} (16y^3 - 13y^2t + 10yt^2 - 3t^3),$$

this extension can be obtained easily.

Requiring 10^{-8} for interval widths, taking $\Lambda = 1$, $\varepsilon = 10^{-18}$, step sizes $h_1 = h_2 = h_3 = 0.081746227283888863$ and $h_4^{(0)} = h_3$ as the initial approximation of step size for the first integration step, after 2.295 s, we have obtained the results presented in Table 6. Figure 2 shows the step size changes. Note that we are able to execute 464 integration steps and obtain the last interval for $t \approx 1.5476$, but step sizes are small (and tend to zero) for t approximately greater than 1.54. If we assume $\text{eps} = 10^{-12}$, then we achieve $t \approx 1.3991$ (smaller t) after 4.758 s, and for $\text{eps} = 10^{-4}$ we can achieve $t \approx 2.0843$ (greater t) after 3.333 s (see Tables 7 and 8 for details).

Example 4 Finally, consider the motion of a simple pendulum described by the equation

$$\varphi'' + u^2 \sin \varphi = 0, \quad (48)$$

where $\varphi = \varphi(t)$, $u = \sqrt{g/L}$, and where g is the gravitational acceleration at the Earth surface and L denotes the pendulum length. If we assume that the angle φ is small, i.e., $\sin \varphi \approx \varphi$, then the equation (48) can be reduced to the equation of simple

Table 6 Intervals obtained for the problem (47) including the last calculated integration step by the method (19) with matching step sizes ($\text{eps} = 10^{-8}$)

k	t_k	Y_k	Width
50	≈ 0.69774	[4.5975465221505685E+0000, 4.5975465243497029E+0000]	$\approx 2.20 \cdot 10^{-9}$
100	≈ 1.14965	[4.9030146239176995E+0000, 4.9030146292963836E+0000]	$\approx 5.38 \cdot 10^{-9}$
150	≈ 1.47461	[5.0930266442612574E+0000, 5.0930266533191654E+0000]	$\approx 9.06 \cdot 10^{-9}$
200	≈ 1.54340	[5.1304617719687623E+0000, 5.1304617819135613E+0000]	$\approx 9.94 \cdot 10^{-9}$
250	≈ 1.54735	[5.1325793100365724E+0000, 5.1325793200333748E+0000]	$< 1.00 \cdot 10^{-8}$
464	≈ 1.54759	[5.1327090107108448E+0000, 5.1327090207108449E+0000]	$< 1.00 \cdot 10^{-8}$

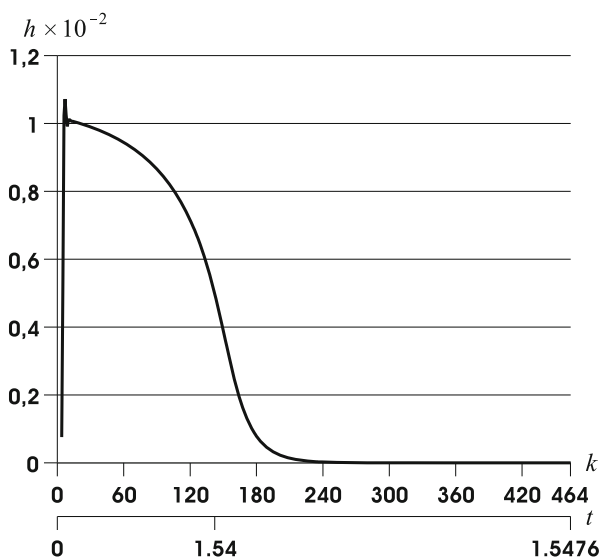


Fig. 2 Step size changes in problem (47) for the method (19)

Table 7 Intervals obtained for the problem (47) with $\epsilon_{ps} = 10^{-12}$

k	t_k	Y_k	Width
200	≈ 0.55405	[4.4886273748015889E+0000, 4.4886273748017744E+0000]	$\approx 1.85 \cdot 10^{-13}$
400	≈ 0.85821	[4.7120930249268474E+0000, 4.712093024927235E+0000]	$\approx 3.76 \cdot 10^{-13}$
600	≈ 1.13809	[4.8958332122088699E+0000, 4.8958332122095143E+0000]	$\approx 6.44 \cdot 10^{-13}$
800	≈ 1.36239	[5.0299271956108516E+0000, 5.0299271956117965E+0000]	$\approx 9.45 \cdot 10^{-13}$
900	≈ 1.39889	[5.0507330026032329E+0000, 5.0507330026042326E+0000]	$< 1.00 \cdot 10^{-12}$
955	≈ 1.39915	[5.0508805112730600E+0000, 5.0508805112740601E+0000]	$< 1.00 \cdot 10^{-12}$

Table 8 Intervals obtained for the problem (47) with $\epsilon_{ps} = 10^{-4}$

k	t_k	Y_k	Width
10	≈ 0.67182	[4.5783575612188736E+0000, 4.5783702917087973E+0000]	$\approx 1.27 \cdot 10^{-5}$
20	≈ 1.22301	[4.9478886689807839E+0000, 4.9479228693528902E+0000]	$\approx 3.42 \cdot 10^{-5}$
30	≈ 1.62289	[5.1725533292643105E+0000, 5.1726121849592280E+0000]	$\approx 5.89 \cdot 10^{-5}$
100	≈ 2.07277	[5.3895608440740060E+0000, 5.3896595445228829E+0000]	$\approx 9.87 \cdot 10^{-5}$
150	≈ 2.08316	[5.3941735254856961E+0000, 5.3942734009330229E+0000]	$\approx 9.99 \cdot 10^{-5}$
632	≈ 2.08426	[5.3946582071369410E+0000, 5.3947582071369411E+0000]	$< 1.00 \cdot 10^{-4}$

Table 9 Starting values for the method (18) and the problem (50)–(51)

k	t_k	$Y_s = Y_{sk}$
0	0	$Y_1 = [\text{0.0000000000000000E} + 0000, \text{0.0000000000000000E} + 0000]$ $Y_2 = [\text{5.2359877559829887E} - 0001, \text{5.2359877559829888E} - 0001]$
1	0.0001	$Y_1 = [-5.1347498487965657E - 0004, -5.1347498487965656E - 0004]$ $Y_2 = [\text{5.2359874992454941E} - 0001, \text{5.2359874992454942E} - 0001]$
2	0.0002	$Y_1 = [-1.0269499194046190E - 0003, -1.0269499194046189E - 0003]$ $Y_2 = [\text{5.2359867290330357E} - 0001, \text{5.2359867290330358E} - 0001]$

harmonic motion

$$\varphi'' + u^2 \varphi = 0 \quad (49)$$

with the solution $\varphi = \varphi_0 \cos(ut)$, where φ_0 is an initial angle. Denoting $y_1 = \varphi'$, $y_2 = \varphi$ and assuming that $\varphi'(0) = 0$, $\varphi(0) = \varphi_0$, we can transform (49) into the following systems of differential equations of first order:

$$y_1' = -u^2 y_2, \quad y_2' = y_1, \quad (50)$$

with initial conditions

$$y_1(0) = 0, \quad y_2(0) = \varphi_0. \quad (51)$$

Let us take $g = 9.80665$, $L = 1$ and $\varphi_0 = \pi/6$. Let

$$\Delta_t = \{t \in \mathbb{R} : 0 \leq t \leq 2\},$$

$$\Delta_y = \{(y_1, y_2) \in \mathbb{R}^2 : \underline{-1.8} \leq y_1 \leq \overline{1.8}, \underline{-0.6} \leq y_2 \leq \overline{0.6}\}$$

and consider the method (18). Assuming $\epsilon ps = 10^{-8}$, taking $\Lambda = 9.80665\pi/6$, $\varepsilon = 10^{-18}$, step sizes $h_1 = h_2 = 0.0001$, $h_3^{(0)} = h_2$ as the initial approximation of step size for the first integration step, and starting points given in Table 9, we have obtained the results presented in Table 10. The CPU time has been equal to

Table 10 Intervals obtained for the problem (50)–(51) including the last calculated integration step by the method (18) with matching step sizes ($\epsilon ps = 10^{-8}$)

k	t_k	Y_k	Width
20	≈ 0.052266	$Y_1 = [-2.6717739871166114E-0001, -2.6717739518323742E-0001]$ $Y_2 = [\text{5.1660096723952543E} - 0001, \text{5.1660096833861906E} - 0001]$	$\approx 3.53 \cdot 10^{-9}$ $\approx 1.10 \cdot 10^{-9}$
40	≈ 0.098329	$Y_1 = [-4.9695523554187293E-0001, -4.9695522814330236E-0001]$ $Y_2 = [\text{4.9897124759486593E} - 0001, \text{4.9897124992870939E} - 0001]$	$\approx 7.40 \cdot 10^{-9}$ $\approx 2.33 \cdot 10^{-9}$
60	≈ 0.123971	$Y_1 = [-6.2069000044409622E-0001, -6.2068999060901059E-0001]$ $Y_2 = [\text{4.8463438814236487E} - 0001, \text{4.8463439125965848E} - 0001]$	$\approx 9.84 \cdot 10^{-9}$ $\approx 3.12 \cdot 10^{-9}$
80	≈ 0.125890	$Y_1 = [-6.2979775595258544E-0001, -6.297977459297822E-0001]$ $Y_2 = [\text{4.8343471268438440E} - 0001, \text{4.8343471585460307E} - 0001]$	$< 1.00 \cdot 10^{-8}$ $\approx 3.17 \cdot 10^{-9}$
122	≈ 0.125895	$Y_1 = [-6.2982173628611325E-0001, -6.2982172628611324E-0001]$ $Y_2 = [\text{4.8343152696498447E} - 0001, \text{4.8343153013532947E} - 0001]$	$< 1.00 \cdot 10^{-8}$ $\approx 3.17 \cdot 10^{-9}$

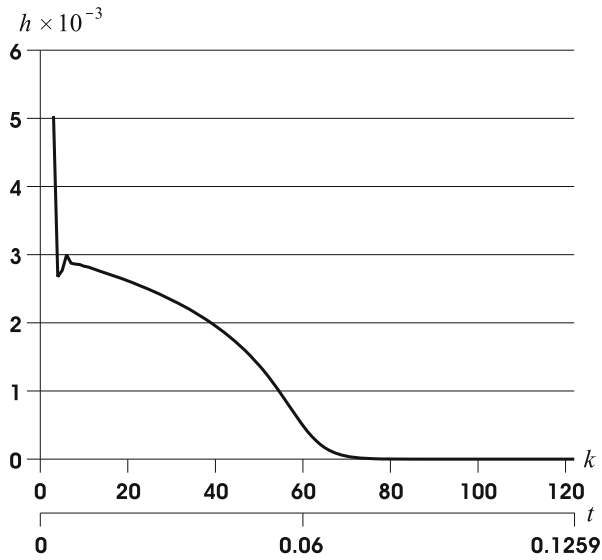


Fig. 3 Step size changes in problem (50)–(51) for the method (18)

0.740 s. The changes of step size are shown in Fig. 3, from which it follows that for t approximately greater than 0.125 the step size is too small (tends to zero) to satisfactory continue the calculations. On the other hand, one can observe that—according to Theorem 1—the exact solution belongs to interval enclosures obtained.

Table 11 CPU times for the method (19) and different values of eps

eps	k	t_{\max}	CPU time (s)
10^{-2}	842	≈ 2.612638	4.411
10^{-3}	731	≈ 2.341260	3.751
10^{-4}	632	≈ 2.084259	3.333
10^{-5}	557	≈ 1.880208	2.816
10^{-6}	503	≈ 1.731451	2.502
10^{-7}	472	≈ 1.624865	2.352
10^{-8}	464	≈ 1.547588	2.295
10^{-9}	489	≈ 1.490991	2.433
10^{-10}	559	≈ 1.450758	2.704
10^{-11}	705	≈ 1.423210	3.496
10^{-12}	955	≈ 1.399150	4.758
10^{-13}	1312	≈ 1.321706	6.575

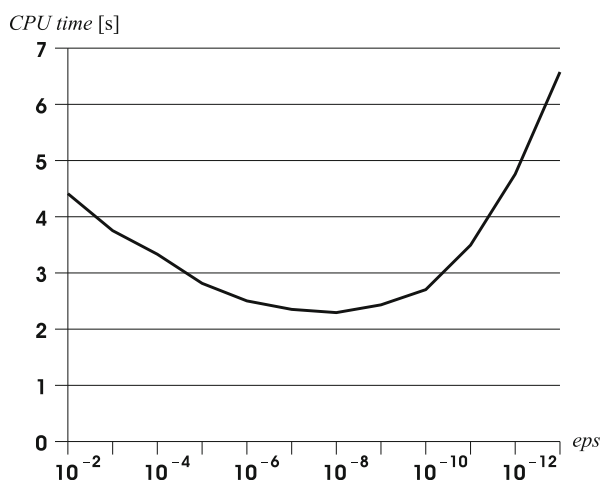


Fig. 4 CPU times for the method (19) and different values of eps

The examples show that using the procedure for step size changing we can obtain interval enclosures of the exact solution with widths given beforehand for $t_{\max} \leq a$. Unfortunately, very often we have $t_{\max} < a$. Of course, to reach a , we can always use a multistep method with constant step size and theorems similar to the Theorem 1 guarantee that the exact solution will be inside interval enclosures obtained.⁵ We can also use a combination of methods with variable and constant step sizes. If we assume that the step size h cannot be less than some h_{\min} , we can start calculations using a method which matches step sizes with a width of intervals given beforehand, and when the step size will be decreased to h_{\min} —continue calculations with $h = h_{\min}$. Of course, such a procedure does not guarantee that at $t \approx a$ we will obtain intervals with a given width, but it can significantly reduce the number of integration steps.

Some remarks concerning computing time for our methods should be given at the end. First of all, let us note that the Adams-Bashforth methods with step size changes are very fast (usually in the process (44), only a few iterations are needed to achieve a desired width eps of interval). Obviously, for a greater n (the number of method steps and the order of method), the computing time is less for a given $t \leq a$. Moreover, for decreasing desired widths of intervals, we can observe decreasing values of $t_{\max} \leq a$, but computing times may not be decreasing (compare the values of t_{\max} and computing times in Example 3 for $eps = 10^{-4}$, 10^{-8} and 10^{-12}). To confirm this observation, let us consider

⁵For a number of interval multistep methods, such theorems have been presented in [9, 17–19, 22, 25–29, 37].

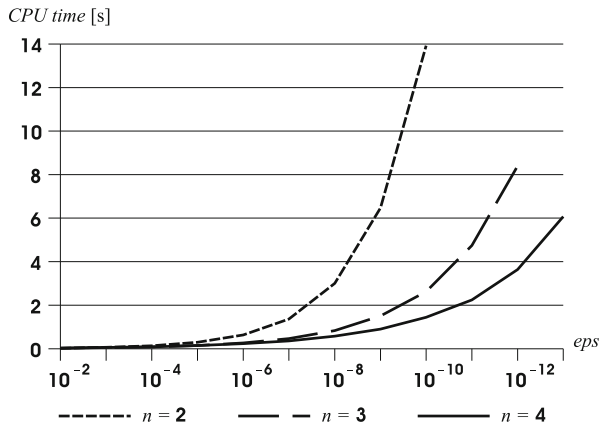


Fig. 5 CPU times needed to achieve $t = 1.3$ for the methods (17), (18), and (19) and different values of eps

Example 5 For the same problem as in Example 3 and for the same Δ_t and Δ_y , let us take different values of eps (from 10^{-2} to 10^{-13}) in the method (19). In Table 11, we present the number of steps k and CPU times to achieve t_{\max} for each value of eps . These times are also presented in Fig. 4, and we can see that for $eps = 10^{-8}$ the computing time is the smallest. It may be generalized: for each problem considered, there exists the smallest computing time for some eps .

On the other hand, for some $t < T \leq a$, where T denotes the smallest t_{\max} common for different n and eps , the smaller computing times are obtained for greater values of n . For the problem considered, the methods (17) ($n = 2$), (18) ($n = 3$), and (19) ($n = 4$), and $t = 1.3$ are shown in Fig. 5.

7 Conclusions

Until now, interval multistep methods (explicit and implicit) have been used only with a constant step size. Using variable step sizes and the procedure presented in Section 4 we can obtain interval enclosure of the exact solution with a width given beforehand. Although we have presented our algorithm only for interval methods of Adams-Bashforth type, it seems that the same procedure one can apply to other multistep methods, including interval predictor-corrector ones [27].

Funding information The paper was financially supported by the Poznan University of Technology (Poland) through the Grants Nos. 09/91/DSPB/1649 and 02/21/DSPB/3544.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Berz, M., Hoffstätter, G.: Computation and application of Taylor polynomials with interval remainder bounds. *Reliab. Comput.* **4**(1), 83–97 (1998)
- Berz, M., Makino, K.: Performance of Taylor model methods for validated integration of ODEs. In: Dongarra, J., Madsen, K., Wasniewski, J. (eds.) *Applied Parallel Computing. State of the Art in Scientific Computing, Lecture Notes in Computer Science*, vol. 3732, pp. 65–73 (2005)
- Butcher, J.C.: *The Numerical Analysis of Ordinary Differential Equations: Runge-Kutta and General Linear Methods*. Wiley, Chichester (1987)
- Ceschino, F.: Modification de la Longueur du pas Dans L'intégration Numérique par les Méthodes à pas Liés. *Chiffres* **2**, 101–106 (1961)
- Collatz, L. *The Numerical Treatment of Differential Equations*, 3rd edn. Springer, Berlin (1959)
- Corliss, G.F., Rihm, R.: Validating an a priori enclosure using high-order Taylor series. In: *Scientific Computing, Computer Arithmetic, and Validated Numerics*, pp. 228–238. Akademie Verlag (1996)
- Enright, W.H., Pryce, H.D.: Two FORTRAN packages for assessing initial value methods. *ACM Trans. Math. Softw.* **13**(1), 1–27 (1987)
- Forrington, C.V.D.: Extension of the predictor–corrector method for the solution of systems of ordinary differential equations. *Comput. J.* **4**(1), 80–84 (1961)
- Gajda, K., Jankowska, M., Marciniak, A., Szyszka, B.: A survey of interval Runge-Kutta and multi-step methods for solving the initial value problem. In: Wyrzykowski, R., Dongarra, J., Karczewski, K., Wasniewski, J. (eds.) *Parallel Processing and Applied Mathematics, Lecture Notes in Computer Science*, vol. 4967, pp. 1361–1371. Springer, Berlin (2008)
- Gajda, K., Marciniak, A., Szyszka, B.: Three-and four-stage implicit interval methods of Runge-Kutta type. *Comput. Methods Sci. Technol.* **6**(1), 41–59 (2000)
- Hairer, E., Nørsett, S.P., Wanner, G.: *Solving Ordinary Differential Equations I – Nonstiff Problems*. Springer, Berlin (1987)
- Hansen, E.R.: *Topics in Interval Analysis*. Oxford University Press, London (1969)
- Henrici, P.: *Discrete Variable Methods in Ordinary Differential Equations*. Wiley, New York (1962)
- Henrici, P.: *Error Propagation in Difference Methods*. Wiley, New York (1963)
- Jackson, K.R., Nedialkov, N.S.: Some recent advances in validated methods for IVPs for ODEs. *Appl. Numer. Math.* **42**(1–3), 269–284 (2002)
- Jain, M.K.: *Numerical Solution of Differential Equations*. Wiley, New York (1979)
- Jankowska, M., Marciniak, A.: Implicit interval methods for solving the initial value problem. *Comput. Methods Sci. Technol.* **8**(1), 17–30 (2002)
- Jankowska, M., Marciniak, A.: On explicit interval methods of Adams-Bashforth type. *Comput. Methods Sci. Technol.* **8**(2), 46–57 (2002)
- Jankowska, M., Marciniak, A.: On two families of implicit interval methods of Adams-Moulton type. *Comput. Methods Sci. Technol.* **12**(2), 109–113 (2006)
- Krogh, F.T.: A variable step variable order multistep method for the numerical solution of ordinary differential equations. In: *Information Processing 68. Proceedings of IFIP Congress 1968*, pp. 194–199. North-Holland, Amsterdam (1968)
- Krogh, F.T.: Changing step size in the integration of differential equations using modified divided differences. In: *Proceedings of the Conference on the Numerical Solutions of ODE. Lecture Notes in Mathematics*, 362, pp. 22–71. Springer-Verlag, New York (1974)
- Marciniak, A.: *Selected Interval Methods for Solving the Initial Value Problem*. Publishing House of Poznan University of Technology, Poznan (2009). <http://www.cs.put.poznan.pl/amarciniak/IMforIVP-book/IMforIVP.pdf>
- Marciniak, A.: Interval Arithmetic Unit. <http://www.cs.put.poznan.pl/amarciniak/IAUnits/IntervalArithmetic32and64.pas> (2016)
- Marciniak, A.: Delphi Pascal programs for step size control in interval Adams–Bashforth methods. <http://www.cs.put.poznan.pl/amarciniak/VSSABM-Examples> (2019)
- Marciniak, A., Jankowska, M.A.: Interval versions for special kinds of explicit linear multistep methods (in review, available from the authors)
- Marciniak, A., Jankowska, M.A.: Interval versions of Milne's multistep method. *Numer. Algorithms* **79**(1), 87–105 (2018)
- Marciniak, A., Jankowska, M.A., Hoffmann, T.: On interval predictor–corrector methods. *Numer. Algorithms* **75**(3), 777–808 (2017)

28. Marciniak, A., Szyszka, B.: One-and two-stage implicit interval methods of Runge-Kutta type. *Comput. Methods Sci. Technol.* **5**(1), 53–65 (1999)
29. Marciniak, A., Szyszka, B.: Interval Runge–Kutta methods with variable step sizes. *Comput. Methods Sci. Technol.* **25**(1), 33–46 (2019)
30. Marciniak, A., Szyszka, B., Hoffmann, T.: An interval version of Kuntzmann–Butcher method for solving the initial value problem (in review, available from the authors)
31. Moore, R.E.: *Interval Analysis*. Prentice-Hall, Englewood Cliffs (1966)
32. Moore, R.E.: *Methods and applications of interval analysis*. SIAM Society for Industrial & Applied Mathematics Philadelphia (1979)
33. Nedialkov, N.S.: VNODE-LP – a validated solver for initial value problems in ordinary differential equations. Tech. Rep. CAS 06-06-NN, Department of Computing and Software, McMaster University, Hamilton (2006)
34. Nedialkov, N.S., Jackson, K.R., Corliss, G.F.: Validated solutions of initial value problems for ordinary differential equations. *Appl. Math. Comput.* **105**(1), 21–68 (1999)
35. Shampine, L.F.: Variable order Adams codes. *Comput. Math. Appl.* **44**(5–6), 749–761 (2002)
36. Shampine, L.F., Gordon, M.K.: *Computer solution of ordinary differential equations* freeman (1975)
37. Shokin, Y.I.: *Interval Analysis*. Novosibirsk, Nauka (1981)
38. Stoer, J., Bulirsch, R.: *Introduction to Numerical Analysis*. Springer, Berlin (1983)

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.