Numerical Solution and Stability of Block Method for Solving Functional Differential Equations

Fuziyah Ishak, Mohamed B. Suleiman and Zanariah A. Majid

Abstract In this article, we describe the development of a two-point block method for solving functional differential equations. The block method, implemented in variable stepsize technique produces two approximations simultaneously using the same back values. The grid-point formulae for the variable steps are derived, calculated and stored at the start of the program for greater efficiency. The delay solutions for the unknown function and its derivative at earlier times are interpolated using the previous computed values. Stability regions for the block method are illustrated. Numerical results are given to demonstrate the accuracy and efficiency of the block method.

Keywords Block method · Delay differential equation · Functional differential equation · Neutral delay differential equation · Polynomial interpolation · Stability region

F. Ishak (✉)
Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA,
40450 Shah Alam, Selangor, Malaysia
e-mail: fuziyah@tmsk.uitm.edu.my

M. B. Suleiman
Institute for Mathematical Research (INSPEM), Universiti Putra Malaysia,
43400 Serdang, Selangor, Malaysia
e-mail: mohamed@math.upm.edu.my

Z. A. Majid
Mathematics Department, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia
e-mail: zanariah@math.upm.edu.my

G.-C. Yang et al. (eds.), *Transactions on Engineering Technologies*,
DOI: 10.1007/978-94-017-8832-8_43,

1 Introduction

Differential equation of the form

$$y'(x) = f(x, y(x), y(\alpha(x)), y'(\alpha(x))) \tag{1}$$

appears in many real life applications and has been investigated by many authors in recent years. The classical case is when $\alpha(x) = x - \tau$, τ a constant. When the right hand side of (1) does not depend on the derivative of the unknown function y, the equation is known as delay differential equation. Otherwise, it is known as neutral delay differential equation. In this article, we consider numerical solution for functional differential equation of the form:

$$\begin{cases} y'(x) = f(x, y(x), y(qx), y'(qx)), & 0 < x \leq T, \\ y(0) = y_0, \end{cases} \tag{2}$$

where $0 < q < 1$, and

$$\begin{cases} y'(x) = f(x, y(x), y(x - \tau), y'(x - \tau)), & 0 < x \leq T, \\ y(x) = \phi(x), & x \leq 0. \end{cases} \tag{3}$$

Equation (2), known as the pantograph equation arises in many physical applications such as number theory, electrodynamics, astrophysics, etc. Detailed explanations can be found in [1–3]. Numerical solutions for (2) and (3) have been studied extensively, see for example [4–11] and the references cited therein. These methods produce one approximation in a single integration step. Block methods, however produce more than one approximation in a step. Block methods have been used to solve wide range of ordinary differential equations as well as delay differential equations (see [12–16] and the references cited therein).

The functional differential equations are solved using a two-point block method in variable step. In a single integration step, two new approximates for the unknown function are obtained using the same stepsize. New approximates for the next block are obtained by keeping the stepsize constant, doubled or halved depending upon the local approximation error. In any variable stepsize method, the coefficients of the method need to be recalculated whenever the stepsize changes. In order to avoid the tedious calculation, the coefficients based on the stepsize ratio are calculated beforehand and stored at the start of the program.

The organization of this article is as follows. In Sect. 2, we briefly describe the development of the variable step block method. Stability region for the block method is discussed in Sect. 3. Numerical results for some functional differential equations are presented in Sect. 4 and finally Sect. 5 is the conclusion.

2 Method Development

Referring to (1), we seek a set of discrete solutions for the unknown function y in the interval [0, T]. The interval is divided into a sequence of mesh points $\{x_i\}_{i=0}^{t}$ of different lengths, such that $0 = x_0 < x_1 < \cdots < x_t = T$. Let the approximated solution for $y(x_n)$ be denoted as y_n. Suppose that the solutions have been obtained up to x_n. At the current step, two new solutions y_{n+1} and y_{n+2} at x_{n+1} and x_{n+2} respectively are simultaneously approximated using the same back values by taking the same stepsize. The points x_{n+1} and x_{n+2} are contained in the current block. The length of the current block is $2h$. We refer to this particular block method as two-point one-block method. The block method is shown in Fig. 1.

In Fig. 1, the stepsize of the previous step is viewed in the multiple of the current stepsize. Thus, $x_{n+1} - x_n = h$, $x_{n+2} - x_{n+1} = h$ and $x_{n\text{-}1} - x_{n-2} = x_n - x_{n-1} = rh$. The value of r is either 1, 2, or $\frac{1}{2}$, depending upon the decision to change the stepsize. In this algorithm, we employ the strategy of having the stepsize to be constant, halved or doubled.

The formulae for the block method can be written as the pair,

$$\begin{aligned} y_{n+1} &= y_n + h\sum_{i=0}^{4} \beta_i(r) f(x_{n-2+i}, y_{n-2+i}, \bar{y}_{n-2+i}, \hat{y}_{n-2+i}), \\ y_{n+2} &= y_n + h\sum_{i=0}^{4} \beta_i^*(r) f(x_{n-2+i}, y_{n-2+i}, \bar{y}_{n-2+i}, \hat{y}_{n-2+i}), \end{aligned} \tag{4}$$

where $\bar{y}_n$ and $\hat{y}_n$ are the approximations to $y(\alpha(x_n))$ and $y'(\alpha(x_n))$ respectively. For simplicity, from now on we refer to $f(x_n, y_n, \bar{y}_n, \hat{y}_n)$ as f_n. The coefficient functions $\beta_i(r)$ and $\beta_i^*(r)$ will give the coefficients of the method when r is either 1, 2, or $\frac{1}{2}$.

The first formula in (4) is obtained by integrating (1) from x_n to x_{n+1} while replacing the function f with the polynomial P where $P(x)$ is given by

$$P(x) = \sum_{j=0}^{4} L_{4,j}(x) f_{n+2-j},$$

and

$$L_{4,j}(x) = \prod_{\substack{i=0 \\ i \neq j}}^{4} \frac{(x - x_{n+2-i})}{(x_{n+2-j} - x_{n+2-i})}, \quad \text{for} \quad j = 0, 1, \ldots, 4.$$

Similarly, the second formula in (4) is obtained by integrating (1) from x_n to x_{n+2} while replacing the function f with the polynomial P. The value of $\bar{y}_n$ is obtained by the interpolation function such as,

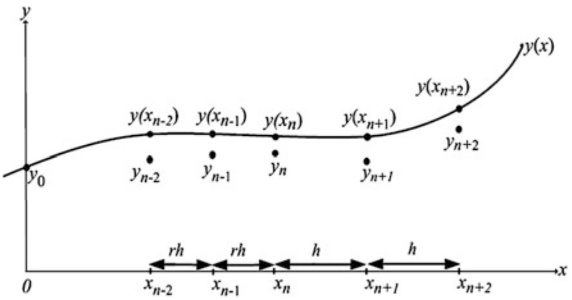

Fig. 1 Two-point one-block method

$$
\begin{aligned}
\bar{y}_n = {} & y[x_j] + (\alpha(x_n) - x_j)y[x_j, x_{j-1}] + \cdots \\
& + (\alpha(x_n) - x_j)\cdots(\alpha(x_n) - x_{j-3})y\,[x_j, \ldots, x_{j-4}],
\end{aligned}
$$

where

$$
y[x_j, x_{j-1}, \ldots, x_{j-4}] = \frac{y[x_j, \ldots, x_{j-3}] - y[x_{j-1}, \ldots, x_{j-4}]}{x_j - x_{j-4}},
$$

provided that $x_{j-1} \leq \alpha(x_n) \leq x_j$, $n \geq j$, $j \geq 1$. We approximate the value of $\hat{y}_n$ by interpolating the values of f, that is,

$$
\begin{aligned}
\hat{y}_n = {} & f[x_j] + (\alpha(x_n) - x_j)f[x_j, x_{j-1}] + \cdots \\
& + (\alpha(x_n) - x_j)\cdots(\alpha(x_n) - x_{j-3})f[x_j, \ldots, x_{j-4}],
\end{aligned}
$$

where

$$
f[x_j, x_{j-1}, \ldots, x_{j-4}] = \frac{f[x_j, \ldots, x_{j-3}] - f[x_{j-1}, \ldots, x_{j-4}]}{x_j - x_{j-4}}.
$$

The formulae in (4) are implicit, thus a set of predictors are derived similarly using the same number of back values. The corrector formulae in (4) are iterated until convergence.

For greater efficiency while achieving the required accuracy, the algorithm is implemented in variable stepsize scheme. The stepsize is changed based on the local error that is controlled at the second point. A step is considered successful if the local error is less than a specified tolerance. If the current step is successful, we consider either doubling or keeping the same stepsize. If the same stepsize had been used for at least two blocks, we double the next stepsize. Otherwise, the next stepsize is kept the same. If the current step fails, the next stepsize is reduced by half. For repeated failures, a restart with the most optimal stepsize with one back value is required. For variable step algorithms, the coefficients of the methods need to be recalculated whenever a stepsize changes. The recalculation cost of these coefficients is avoided by calculating the coefficients beforehand and storing them at the start of the program. With our stepsize changing strategy, we store the coefficients $\beta_i(r)$ and $\beta_i^*(r)$ for r is 1, 2 and $\frac{1}{2}$.

3 Region of Absolute Stability

In the development of a numerical method, it is of practical importance to study the behavior of the global error. The numerical solution y_n is expected to behave as the exact solution $y(x_n)$ does as x_n approaches infinity. In this section, we present the result of stability analysis of the two-point one-block method when they are applied to the delay and neutral delay differential equations with real coefficients.

For the sake of simplicity and without the lost of generality, we consider the equation

$$\begin{aligned} y'(x) &= ay(x) + by(x-\tau) + cy'(x-\tau), & x \geq 0, \\ y(x) &= \phi(x), & -\tau \leq x < 0, \end{aligned} \tag{5}$$

where a, b, $c \in R$, τ is the delay term such as $\tau = mh$, h is a constant stepsize such that $x_n = x_0 + nh$ and $m \in Z^+$. If $i \in Z^+$, we define vectors $\mathbf{Y}_{N+i} = \begin{bmatrix} y_{n-3+2i} \\ y_{n-2+2i} \end{bmatrix}$ and $\mathbf{F}_{N+i} = \begin{bmatrix} f_{n-3+2i} \\ f_{n-2+2i} \end{bmatrix}$. Then, the block method (4) can be written in matrix form such as,

$$A_1\mathbf{Y}_{N+1} + A_2\mathbf{Y}_{N+2} = h\sum_{i=0}^{2} B_i(r)\mathbf{F}_{N+i}, \tag{6}$$

where $A_1 = \begin{bmatrix} 0 & -1 \\ 0 & -1 \end{bmatrix}$, $A_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, and $B_i(r)$ is a matrix that contains the coefficients $\beta_i(r)$ and $\beta_i^*(r)$. Applying method (6) to (5), we get

$$\begin{aligned} A_1\mathbf{Y}_{N+1} + A_2\mathbf{Y}_{N+2} =& H_1\sum_{i=0}^{2} B_i(r)\mathbf{Y}_{N+i} + H_2\sum_{i=0}^{2} B_i(r)\mathbf{Y}_{N+i-m} \\ &+ cA_2\mathbf{Y}_{N+2-m} + cA_1\mathbf{Y}_{N+1-m}, \end{aligned}$$

where $H_1 = ha$ and $H_2 = hb$. Rearranging, we have

$$\sum_{i=0}^{2} (A_i - H_1B_i(r))\mathbf{Y}_{N+i} = \sum_{i=0}^{2} (H_2B_i(r) + cA_i)\mathbf{Y}_{N+i-m}, \tag{7}$$

where A_0 is the null matrix. Characteristic polynomial for (7) is given by $C_m(H_1, H_2, c; \zeta)$ where C_m is the determinant of

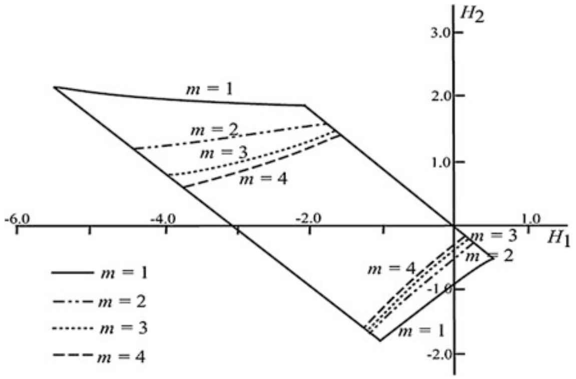

Fig. 2 Stability regions for the block method with $c = 0$, $r = 1$

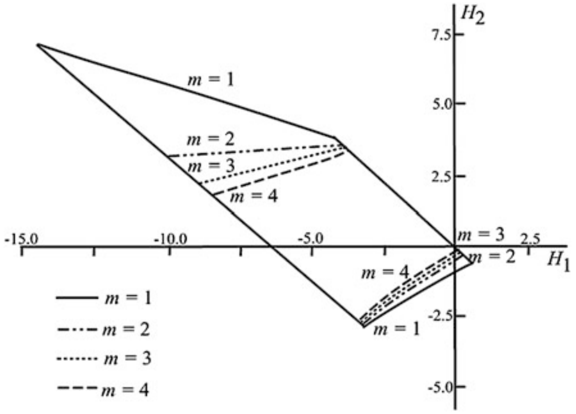

Fig. 3 Stability regions for the block method with $c = 0$, $r = 2$

$$\sum_{i=0}^{2}(A_i - H_1B_i(r))\zeta^{m+i} - \sum_{i=0}^{2}(H_2B_i(r) + cA_i)\zeta^i = 0. \tag{8}$$

The numerical solution (7) is asymptotically stable if and only if for all m, all zeros of the characteristic polynomial (8) lie within the open unit disk in the plane. The stability region is defined as follows:

Definition 1 For a fixed stepsize h, a, $b \in R$, and for any, but fixed c, the region S in the $H_1 - H_2$ plane is called the stability region of the method if for any $(H_1, H_2) \in S$, the numerical solution of (5) vanishes as x_n approaches infinity.

In Figs. 2, 3, 4 the stability regions for $c = 0$ are depicted with $r = 1$, $r = 2$, and $r = \frac{1}{2}$, respectively, see also [15]. In Fig. 5, the stability regions for $m = 1$ and $c = 0.5$ are illustrated. We use the boundary locus technique as described in [17, 18]. The regions are sketched for $r = 1$, $r = 2$, and $r = \frac{1}{2}$.

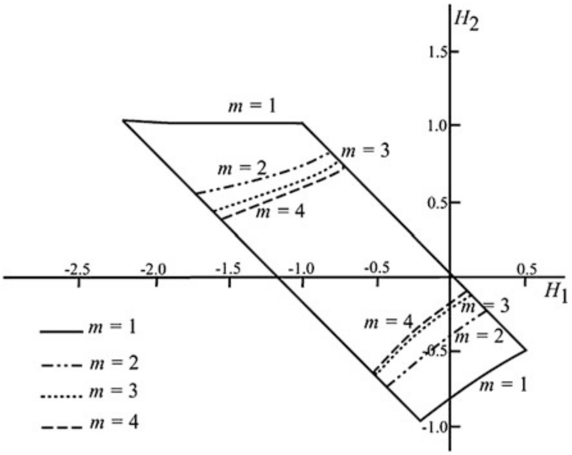

Fig. 4 Stability regions for the block method with $c = 0$, $r = 0.5$

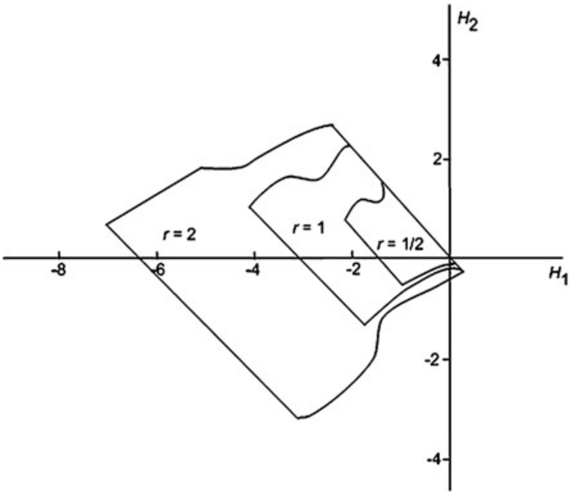

Fig. 5 Stability regions for the block method with $c = 0.5$

The coefficient matrices are given as follows:
For $r = 1$:

$$B_0 = \begin{bmatrix} 0 & \frac{11}{720} \\ 0 & \frac{-1}{90} \end{bmatrix}, \quad B_1 = \begin{bmatrix} \frac{-74}{720} & \frac{456}{720} \\ \frac{4}{90} & \frac{24}{90} \end{bmatrix}, \quad \text{and} \quad B_2 = \begin{bmatrix} \frac{346}{720} & \frac{-10}{720} \\ \frac{124}{90} & \frac{29}{90} \end{bmatrix}.$$

For $r = 2$:

$$B_0 = \begin{bmatrix} 0 & \frac{137}{14400} \\ 0 & \frac{-1}{900} \end{bmatrix}, \quad B_1 = \begin{bmatrix} \frac{-335}{14400} & \frac{7455}{14400} \\ \frac{5}{900} & \frac{285}{900} \end{bmatrix}, \quad \text{and} \quad B_2 = \begin{bmatrix} \frac{7808}{14400} & \frac{-565}{14400} \\ \frac{1216}{900} & \frac{295}{900} \end{bmatrix}.$$

For $r = \frac{1}{2}$:

$$B_0 = \begin{bmatrix} 0 & \frac{145}{1800} \\ 0 & \frac{-20}{225} \end{bmatrix}, \quad B_1 = \begin{bmatrix} \frac{-704}{1800} & \frac{1635}{1800} \\ \frac{64}{225} & \frac{15}{225} \end{bmatrix}, \quad \text{and} \quad B_2 = \begin{bmatrix} \frac{755}{1800} & \frac{-31}{1800} \\ \frac{320}{225} & \frac{71}{225} \end{bmatrix}.$$

Referring to Figs. 2, 3, 4, 5, the stability regions are closed region bounded by the corresponding boundary curves. It is observed that the stability region shrinks as the stepsize increases.

4 Numerical Results

In this section, we present some numerical examples in order to illustrate the accuracy and efficiency of the block method. The examples taken and cited from [8, 19] are as follows:

Example 1

$$\begin{aligned} y'(x) &= \frac{1}{2}y(x) + \frac{1}{2}e^{x/2}y\left(\frac{x}{2}\right), \quad 0 \le x \le 1, \\ y(0) &= 1. \end{aligned}$$

The exact solution is $y(x) = e^x$.

Example 2

$$\begin{aligned} y'(x) &= -\frac{5}{4}e^{-x/4}y\left(\frac{4}{5}x\right), \quad 0 \le x \le 1, \\ y(0) &= 1. \end{aligned}$$

The exact solution is $y(x) = e^{-1.25x}$.

Example 3

$$\begin{aligned} y'(x) &= -y(x) + \frac{q}{2}y(qx) - \frac{q}{2}e^{-qx}, \quad 0 \le x \le 1, \\ y(0) &= 1. \end{aligned}$$

The exact solution is $y(x) = e^{-x}$.

Example 4

$$\begin{aligned} y'(x) &= ay(x) + by(qx) + \cos x - a\sin x - b\sin(qx), \quad 0 \le x \le 1, \\ y(0) &= 0. \end{aligned}$$

The exact solution is $y(x) = \sin x$.

Table 1 Numerical results for Example 1

TOL	STEP	FS	AVERR	MAXE
10^{-2}	20	0	7.02683E−01	9.68012E−05
10^{-4}	27	0	4.15905E−07	9.05425E−07
10^{-6}	35	0	3.09498E−07	4.21310E−07
10^{-8}	48	0	9.55941E−09	1.28084E−08
10^{-10}	75	0	7.68855E−11	9.88663E−11

Table 2 Numerical results for Example 2

TOL	STEP	FS	AVERR	MAXE
10^{-2}	21	0	9.76063E−08	1.17093E−07
10^{-4}	27	0	4.95202E−09	1.27731E−07
10^{-6}	35	0	1.06955E−09	1.21167E−08
10^{-8}	50	0	3.12606E−11	1.43148E−10
10^{-10}	79	0	5.60297E−13	1.51457E−12

Example 5

$$\begin{aligned} y'(x) =& -y(x) + \frac{1}{2}y\left(\frac{x}{2}\right) + \frac{1}{2}y'\left(\frac{x}{2}\right), \quad 0 \le x \le 1, \\ y(0) =& 1. \end{aligned}$$

The exact solution is $y(x) = e^{-x}$.

Example 6

$$\begin{aligned} y'(x) =& -y(x) + 0.1y(0.8x) + 0.5y'(0.8x) \\ & + (0.32x - 0.5)e^{-0.8x} + e^{-x}, \quad 0 \le x \le 10, \\ y(0) =& 0. \end{aligned}$$

The exact solution is $y(x) = xe^{-x}$.

Example 7

$$\begin{aligned} y'(x) &= y(x) + y(x-1) - \tfrac{1}{4}y'(x-1), \quad & 0 \le x \le 1, \\ y(x) &= -x, & x \le 0. \end{aligned}$$

The exact solution is $y(x) = -\frac{1}{4} + x + \frac{1}{4}e^{x}$.

Example 8

$$\begin{aligned} y'(x) &= y(x) + y(x-1) - 2y'(x-1), \quad & 0 \le x \le 1, \\ y(x) &= -x, & x \le 0. \end{aligned}$$

The exact solution is $y(x) = -2 + x + 2e^{x}$.

Numerical results for Example 1–8 are given in Tables 1, 2, 3, 4, 5, 6, 7, 8, 9, 10. The following abbreviations are used in the tables, TOL—the chosen tolerance,

Table 3 Numerical results for Example 3, $q = 0.2$

TOL	STEP	FS	AVERR	MAXE
10^{-2}	20	0	1.54360E−05	1.91824E−05
10^{-4}	27	0	1.15315E−07	5.14934E−07
10^{-6}	35	0	6.83967E−08	8.52745E−08
10^{-8}	47	0	2.03160E−09	2.57931E−09
10^{-10}	74	0	1.50180E−11	2.00260E−11

Table 4 Numerical results for Example 3, $q = 0.8$

TOL	STEP	FS	AVERR	MAXE
10^{-2}	20	0	1.57507E−07	2.30455E−06
10^{-4}	27	0	2.90818E−08	4.90683E−07
10^{-6}	35	0	4.03006E−10	3.90288E−09
10^{-8}	48	0	2.46267E−11	1.51318E−10
10^{-10}	74	0	3.09422E−13	1.74260E−12

Table 5 Numerical results for Example 4, $a = -1, b = 0.5, q = 0.1$

TOL	STEP	FS	AVERR	MAXE
10^{-2}	20	0	1.61996E−06	1.15114E−05
10^{-4}	27	0	1.08039E−07	1.15418E−06
10^{-6}	35	0	8.23406E−09	1.15448E−07
10^{-8}	48	0	5.74175E−10	1.15451E−08
10^{-10}	79	0	4.12730E−11	1.15451E−09

Table 6 Numerical results for Example 4, $a = -1, b = 0.5, q = 0.5$

TOL	STEP	FS	AVERR	MAXE
10^{-2}	20	0	6.25587E−06	4.77895E−05
10^{-4}	27	0	4.53165E−07	4.78257E−06
10^{-6}	35	0	3.48942E−08	4.78294E−07
10^{-8}	50	0	2.91205E−09	4.78298E−08
10^{-10}	79	0	1.83823E−10	4.78299E−09

Table 7 Numerical results for Example 5

TOL	STEP	FS	AVERR	MAXE
10^{-2}	20	0	7.31358E−04	1.83509E−03
10^{-4}	27	0	1.88577E−05	2.92294E−05
10^{-6}	71	0	9.02824E−06	2.12659E−05
10^{-8}	166	2	1.14939E−06	2.13569E−06
10^{-10}	236	4	4.56047E−08	5.25142E−08

Table 8 Numerical results for Example 6

TOL	STEP	FS	AVERR	MAXE
10^{-2}	70	0	1.22749E−02	4.54860E−02
10^{-4}	97	0	3.92866E−04	1.14032E−03
10^{-6}	118	0	1.61615E−06	4.86641E−06
10^{-8}	173	0	2.72657E−07	4.87304E−07
10^{-10}	300	3	1.72160E−08	3.97650E−08

Table 9 Numerical results for Example 7

TOL	STEP	FS	AVERR	MAXE
10^{-2}	21	0	9.06905E−08	8.55589E−07
10^{-4}	27	0	1.68103E−08	3.01837E−07
10^{-6}	34	0	7.73475E−10	1.19985E−08
10^{-8}	43	0	4.71906E−11	5.91029E−10
10^{-10}	60	0	1.41839E−12	9.05609E−12

Table 10 Numerical results for Example 8

TOL	STEP	FS	AVERR	MAXE
10^{-2}	22	0	3.41065E−07	3.95940E−06
10^{-4}	29	0	1.77364E−08	1.88569E−07
10^{-6}	36	0	9.95212E−10	1.52315E−08
10^{-8}	48	0	4.20781E−11	3.12677E−10
10^{-10}	72	0	9.49656E−13	3.90998E−12

STEP—the total number of steps taken, FS—the number of failed steps, AVERR—the average error, and MAXE—the maximum error. The notation 7.02683E−01 means 7.02683×10^{-1}.

From Tables 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, it is observed that for the given tolerances, the two-point block method achieves the desired accuracy. When the tolerance becomes smaller, the total number of steps increases. In order to achieve the desired accuracy, smaller stepsizes are taken, thus resulting in the increase number of total steps taken.

5 Conclusion and Future Work

In this paper, we have discussed the development of a two-point block method for solving functional differential equations of delay and neutral delay-types. The block method produces two approximate solutions in a single integration step by using the same back values. The algorithm is implemented in variable stepsize technique where the coefficients for the various stepsizes are stored at the beginning of the program for greater efficiency. Stability regions for a general linear test equation are obtained for a fixed, but variable stepsizes. The numerical results indicate that the two-point block method achieves the desired accuracy as efficiently as possible.

In the future, the focus for the research should include the implementation of the block method on parallel machines. The efficiency of the block method can be fully utilized if the computation for each point can be divided among parallel tasks.

References

1. R.D. Driver, *Ordinary and Delay Differential Equations* (Springer, New York, 1977)
2. J.R. Ockendon, A.B. Taylor, The dynamics of a current collection system for an electric locomotive. Proc. R. Soc. Lond., Ser. A **322**, 447–468 (1971)
3. A. Iserles, On the generalized pantograph functional differential equation. Eur. J. Appl. Math. **4**, 1–38 (1992)
4. F. Ishak, M.B. Suleiman, Z.A. Majid, Block method for solving pantograph-type functional differential equations, in Lecture Notes in Engineering and Computer Science: Proceedings of the World Congress on Engineering 2013, WCE 2013, 3–5 July 2013, London, UK, pp. 948–952
5. D.J. Evans, K.R. Raslan, The Adomian decomposition method for solving delay differential equation. Int. J. Comput. Math. **82**(1), 49–54 (2005)
6. W.S. Wang, S.F. Li, On the one-leg θ-methods for solving nonlinear neutral functional differential equations. Appl. Math. Comput. **193**, 285–301 (2007)
7. I. Ali, H. Brunner, T. Tang, A spectral method pantograph-type delay differential equations and its convergence analysis. J. Comput. Math. **27**(2–3), 254–265 (2009)
8. S. Sedaghat, Y. Ordokhani, M. Dehghan, Numerical solution of the delay differential equations of pantograph type via Chebyshev polynomials. Commun. Nonlinear Sci. Numer. Simul. **137**, 4815–4830 (2012)
9. F. Samat, F. Ismail, M. Suleiman, Phase fitted and amplification fitted hybrid methods for solving second order ordinary differential equations. IAENG Int. J. Appl. Math. **43**(3), 95–105 (2013)
10. C. Yang, J. Hou, Numerical method for solving volterra integral equations with a convolution kernel. IAENG Int. J. Appl. Math. **43**(4), 185–189 (2013)
11. K.M. Hsiao, W.Y. Lin, F. Fuji, Free vibration analysis of rotating Euler beam by finite element method. Eng. Lett. **20**(3), 253–258 (2012)
12. T.A. Anake, D.O. Awoyemi, A.O. Adesanya, One-step implicit hybrid block method for the direct solution of general second order ordinary differential equations. IAENG Int. J. Appl. Math. **42**(4), 224–228 (2012)
13. Z.A. Majid, Parallel block methods for solving ordinary differential equations. Ph.D. thesis, Universiti Putra Malaysia (2004)
14. F. Ishak, M. Suleiman, Z. Omar, Two-point predictor-corrector block method for solving delay differential equations. Matematika **24**(2), 131–140 (2008)
15. F. Ishak, Z.A. Majid, M. Suleiman, Two-point block method in variable stepsize technique for solving delay differential equations. J. Mater. Sci. Eng. **4**(12), 86–90 (2010)
16. F. Ishak, Z.A. Majid, M.B. Suleiman, Development of implicit block method for solving delay differential equations, in Proceedings of the 14th WSEAS International Conference on Mathematical and Computational Methods in Science and Engineering, Malta, 2012, pp. 67–71
17. C.T.H. Baker, C.A.H. Paul, Computing stability regions-Runge-Kutta methods for delay differential equations. IMA J. Numer. Anal. **14**, 347–362 (1994)

18. A.N. Al-Mutib, Stability properties of numerical methods for solving delay differential equations. J. Comp. Appl. Math. **10**, 71–79 (1984)
19. C.A.H. Paul, A test set of functional differential equations, in Numerical Analysis Report No. 243, February 1994