Sains Malaysiana 48(3)(2019): 677–684
http://dx.doi.org/10.17576/jsm-2019-4803-22

Fifth Order Multistep Block Method for Solving Volterra Integro-Differential Equations of Second Kind

(Kaedah Blok Berbilanglangkah Peringkat Lima bagi Penyelesaian Persamaan Pembezaan - Kamiran Volterra Jenis Kedua)

ZANARIAH ABDUL MAJID* & NURUL ATIKAH MOHAMED

ABSTRACT

In the present paper, the multistep block method is proposed to solve the linear and non-linear Volterra integro-differential equations (VIDEs) of the second kind using constant step size. The proposed block method of order five consists of two point block method presented as in the simple form of Adams Moulton type. The numerical solutions are obtained at two new values simultaneously at each of the integration step. In VIDEs, the unknown function appears in the form of derivative and under the integral sign. The approximation of the integral part is estimated using the Boole's quadrature rule. The stability region is shown, and the numerical results are presented to illustrate the performance of the proposed method in terms of accuracy, total function calls and execution times compared to the existing method.

Keywords: Block method; quadrature rule; Volterra integro-differential equation

ABSTRAK

Dalam makalah ini, kaedah blok berbilanglangkah dicadangkan bagi menyelesaikan persamaan pembezaan-kamiran Volterra (PPKV) linear dan tak linear daripada jenis kedua menggunakan saiz langkah yang malar. Kaedah blok peringkat lima yang dicadangkan terdiri daripada dua titik blok yang dibentangkan dalam bentuk yang mudah daripada jenis Adams Moulton. Penyelesaian berangka diperoleh dalam dua nilai baru pada masa yang sama di setiap langkah kamiran. Dalam PPKV, fungsi yang tidak diketahui muncul dalam bentuk terbitan dan tanda kamiran. Penghampiran bahagian kamiran dianggarkan dengan menggunakan peraturan kuadratur Boole. Rantau kestabilan ditunjukkan dan keputusan berangka dibentangkan untuk menggambarkan prestasi kaedah yang dicadangkan daripada segi kejituan, jumlah panggilan fungsi dan masa pelaksanaan berbanding kaedah sedia ada.

Kata kunci: Aturan kuadratur; kaedah blok; persamaan pembezaan-kamiran Volterra

INTRODUCTION

VIDEs appeared in many physical applications such as in glass forming process, nano hydrodynamics, heat transfer, diffusion process in general and neutron diffusion. The following initial value problems for general Volterra integro-differential equations (VIDEs) will be considered:

$$y'(x) = F(x, y(x), z(x)),\ \ y(0) = y_0,\ \ 0 \le x \le a \qquad (1)$$

$$z(x) = \int_0^x K(x,s,y(s))\,ds \qquad (2)$$

Many different methods have been used to solve the VIDEs problems such as in Chang (1982), Day (1967), Dehghan and Salehi (2012), Filiz (2013, 2014), Ishak and Ahmad (2016); Kürkçü et al. (2017) and Linz (1969). The used of numerical quadrature rules for solving VIDEs has been first discussed by Day (1967). He solved the VIDEs by using the composite trapezoidal rule. Then, Linz (1969) has introduced the combination of linear multistep method and numerical quadrature rules for solving the differential part and integral part of VIDEs. The convergence of such methods has been studied by Linz (1969) and Mocarsky (1971). Chang (1982) has studied the linear multistep method by using two-step and three-step Adams-Moulton method with Euler-Maclaurin for solving VIDEs. Later, Makroglou (1982) has implemented the theory and stability of the hybrid method for the solution of VIDEs. Mohamed and Majid (2016) have introduced multistep block method for solving Volterra integro-differential equation. Recently, Kürkçü et al. (2017) have proposed the collocation method based on residual error analysis for solving integro-differential equations.

An earlier work of one-step algorithms for the numerical solution of VIDEs has been done by Feldstein and Sopka (1974). Then, Runge-Kutta theory for solving VIDEs problem together with its global convergence has been ingeniously studied by Lubich (1982). Yuan and Tang (1990) proposed implicit Runge-Kutta method for solving the nonlinear integro differential equation. In Filiz (2014, 2013), both articles have solved VIDEs using Runge-Kutta method and paired it with Newton Cotes quadrature rule.

In this paper, we present the fifth order multistep block method derived in Majid and Suleiman (2011) with the Boole's quadrature rule for solving linear and nonlinear (1) and (2) of second kind using constant step size.

MATERIALS AND METHODS

The two point three-step block method has been derived earlier by Majid and Suleiman (2011). The derived method based on predictor-corrector pair is used to solve for first order ordinary differential equations (ODEs). The set of points $\{x_{n-3}, x_{n-2}, x_{n-1}, x_n\}$ are used to derive the predictor formulas while the set of points are involved in deriving the corrector formulas. The method is derived using the Lagrange interpolating polynomial.

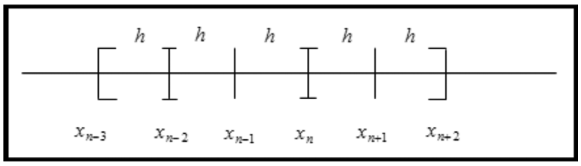

FIGURE 1.Two point three-step block method

In Figure 1, the two point of y_{n+1} and y_{n+2} are obtained by integrating $y' = f(x, y)$ over the interval $[x_n, x_{n+1}]$ and $[x_n, x_{n+2}]$. The predictor formula of the two point three-step block method are derived using Lagrange interpolation polynomial of order four as (3) while the corrector formula of the two point three-step block method are derived using Lagrange interpolation polynomial of order five as (4):

$$
\begin{aligned}
P_4(x) = & \frac{(x-x_{n-3})(x-x_{n-2})(x-x_{n-1})}{(x_n-x_{n-3})(x_n-x_{n-2})(x_n-x_{n-1})}F_n \\
& + \frac{(x-x_{n-3})(x-x_{n-2})(x-x_n)}{(x_{n-1}-x_{n-3})(x_{n-1}-x_{n-2})(x_{n-1}-x_n)}F_{n-1} \\
& + \frac{(x-x_{n-3})(x-x_{n-1})(x-x_n)}{(x_{n-2}-x_{n-3})(x_{n-2}-x_{n-1})(x_{n-1}-x_n)}F_{n-2} \\
& + \frac{(x-x_{n-2})(x-x_{n-1})(x-x_n)}{(x_{n-3}-x_{n-2})(x_{n-3}-x_{n-1})(x_{n-3}-x_n)}F_{n-3}
\end{aligned}
\quad (3)
$$

$$
\begin{aligned}
P_5(x) = & \frac{(x-x_{n-2})(x-x_{n-1})(x-x_n)(x-x_{n+1})}{(x_{n+2}-x_{n-2})(x_{n+2}-x_{n-1})(x_{n+2}-x_n)(x_{n+2}-x_{n+1})}F_{n+2} \\
& + \frac{(x-x_{n-2})(x-x_{n-1})(x-x_n)(x-x_{n+2})}{(x_{n+1}-x_{n-2})(x_{n+1}-x_{n-1})(x_{n+1}-x_n)(x_{n+1}-x_{n+2})}F_{n+1} \\
& + \frac{(x-x_{n-2})(x-x_{n-1})(x-x_{n+1})(x-x_{n+2})}{(x_n-x_{n-2})(x_n-x_{n-1})(x_n-x_{n+1})(x_n-x_{n+2})}F_n \\
& + \frac{(x-x_{n-2})(x-x_n)(x-x_{n+1})(x-x_{n+2})}{(x_{n-1}-x_{n-2})(x_{n-1}-x_n)(x_{n-1}-x_{n+1})(x_{n-1}-x_{n+2})}F_{n-1} \\
& + \frac{(x-x_{n-1})(x-x_n)(x-x_{n+1})(x-x_{n+2})}{(x_{n-2}-x_{n-1})(x_{n-2}-x_n)(x_{n-2}-x_{n+1})(x_{n-2}-x_{n+2})}F_{n-2}
\end{aligned}
\quad (4)
$$

Then, substitute $s = \frac{x - x_{n+2}}{h}$ in (3) and (4), changing the limit of integration and replace $dx = hds$, hence the desired predictor and corrector formulas are obtained as follows:

Predictor formula:

$$
\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\begin{bmatrix} y^p_{n+1} \\ y^p_{n+2} \end{bmatrix}
=
\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}
\begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix}
+ h \begin{bmatrix} -\frac{59}{24} & \frac{55}{24} \\ -\frac{44}{3} & \frac{27}{3} \end{bmatrix}
\begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix}
+ h \begin{bmatrix} -\frac{9}{24} & \frac{37}{24} \\ -\frac{8}{3} & \frac{31}{3} \end{bmatrix}
\begin{bmatrix} f_{n-3} \\ f_{n-2} \end{bmatrix}
\quad (5)
$$

Corrector formula:

$$
\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\begin{bmatrix} y^c_{n+1} \\ y^c_{n+2} \end{bmatrix}
=
\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}
\begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix}
+ h \begin{bmatrix} \frac{346}{720} & -\frac{19}{720} \\ \frac{124}{90} & \frac{29}{90} \end{bmatrix}
\begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix}
+ h \begin{bmatrix} -\frac{74}{720} & \frac{456}{720} \\ \frac{4}{90} & \frac{24}{90} \end{bmatrix}
\begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix}
+ h \begin{bmatrix} 0 & \frac{11}{720} \\ 0 & -\frac{1}{90} \end{bmatrix}
\begin{bmatrix} f_{n-3} \\ f_{n-2} \end{bmatrix}
\quad (6)
$$

The order of the method in (5) and (6) are determined by applying Definition in Lambert (1973): The difference operator L defined by $L[y(x); h] = \sum_{j=0}^{k} [\alpha_j y(x + jh] - h\beta_j y'$ and associated with the linear multistep method (LMM) $\sum_{j=0}^{k} \alpha_j y_{n+j} = h \sum_{j=0}^{k} \beta_j f_{n+j}$ where α_j and β_j are constant. The LMM are said to be of order q if $C_0 = C_1 = \ldots C_q = 0$ and $C_{q+1} \neq 0$. The formula for the constant, C_q defined as,

$$
C_q = \sum_{j=0}^{k} \frac{j^q \alpha_j}{q!} - \frac{j^{q-1}\beta_j}{(q-1)!}. \quad (7)
$$

The predictor formula (5) is implement in (7) and since $C_0 = C_1 = C_2 = C_3 = C_4 = 0$ and $C_5 \neq 0$, hence the method is of order four and the error constant is,

$$C_5 = \begin{pmatrix} 0.34861 \\ 2.98889 \end{pmatrix}. \tag{8}$$

Next, the order of the method in (6) is determined by applying the same formula as in (7). The corrector formula of the two point three-step block method is order five and the error constant is,

$$C_6 = \begin{pmatrix} 0.00764 \\ -0.01111 \end{pmatrix}. \tag{9}$$

The multistep block method for solving linear and nonlinear VIDEs has been written in C language and implemented in the Microsoft Visual C++ environment. The implementation involved the two point three-step block method of order five with Boole's rule for the problems when $K(x, s) = 1$ in (2). The formula for Boole's rule is given as,

$$z_{n+4} = z_n + \frac{2h}{45}\left(7y_{n+4} + 32y_{n+3} + 12y_{n+2} + 32y_{n+1} + 7y_n\right) \tag{10}$$

The composite Boole's rule with interpolation scheme is adapted for solving (1) when $K(x, s) \neq 1$ in (2). Consider the interval [a. b] is subdivided into $4m$ subintervals of equal width $h = \frac{b-a}{4m}$. Hence,

$$\begin{aligned} I &= \int_a^b z(x)\,dx \\ &= \int_{x_0}^{x_n} z(x)\,dx \\ &= \int_{x_0}^{x_4} z(x)\,dx + \int_{x_4}^{x_8} z(x)\,dx + \cdots + \int_{x_{n-4}}^{x_n} z(x)\,dx. \end{aligned}$$

$$I = \frac{2h}{45}\sum_{k=1}^{m}\left(7z\left(x_{4k-4}\right) + 32z\left(x_{4k-3}\right) + 12z\left(x_{4k-2}\right) + 32z\left(x_{4k-1}\right) + 7z\left(x_{4k}\right)\right)$$

Using composite Boole's rule, for $n = 0, 4, 8, \ldots$.

$$z_{n+4} = \frac{2h}{45}\sum_{i=0}^{n+4}\omega_i^s K\left(x_{n+4}, x_i, y_i\right) \tag{11}$$

$$z_{n+5} = \frac{2h}{45}\sum_{i=0}^{n+4}\omega_i^s K\left(x_{n+5}, x_i, y_i\right) + \frac{h}{90}\left\{\begin{aligned} &7K\left(x_{n+5}, x_{n+4}, y_{n+4}\right) \\ &+32K\left(x_{n+5}, x_{n+\frac{17}{4}}, y_{n+\frac{17}{4}}\right) \\ &+12K\left(x_{n+5}, x_{n+\frac{9}{2}}, y_{n+\frac{9}{2}}\right) \\ &+32K\left(x_{n+5}, x_{n+\frac{19}{4}}, y_{n+\frac{19}{4}}\right) \\ &+7K\left(x_{n+5}, x_{n+5}, y_{n+5}\right) \end{aligned}\right\} \tag{12}$$

Lagrange interpolation at points $\{x_{n+1}, x_{n+2}, x_{n+3}, x_{n+4}, x_{n+5}\}$ is used to calculate for unknown values $y_{n+\frac{17}{4}}, y_{n+\frac{9}{2}}, y_{n+\frac{19}{4}}$. The following formulas have been derived:

$$\begin{aligned} y_{n+\frac{17}{4}} &= -\frac{45}{2048}y_{n+1} + \frac{65}{512}y_{n+2} - \frac{351}{1024}y_{n+3} \\ &\quad + \frac{585}{512}y_{n+4} + \frac{195}{2048}y_{n+5} \\ y_{n+\frac{9}{2}} &= -\frac{5}{128}y_{n+1} + \frac{7}{32}y_{n+2} - \frac{35}{64}y_{n+3} \\ &\quad + \frac{35}{32}y_{n+4} + \frac{35}{128}y_{n+5} \\ y_{n+\frac{19}{4}} &= -\frac{77}{2048}y_{n+1} + \frac{105}{512}y_{n+2} - \frac{495}{1024}y_{n+3} \\ &\quad + \frac{385}{512}y_{n+4} + \frac{1155}{2048}y_{n+5} \end{aligned} \tag{13}$$

$$\begin{aligned} z_{n+6} = {} & \frac{2h}{45}\sum_{i=0}^{n+4}\omega_i^s K\left(x_{n+6}, x_i, y_i\right) + \frac{h}{90}\left\{\begin{aligned} &7K\left(x_{n+6}, x_{n+4}, y_{n+4}\right) \\ &+32K\left(x_{n+6}, x_{n+\frac{17}{4}}, y_{n+\frac{17}{4}}\right) \\ &+12K\left(x_{n+6}, x_{n+\frac{9}{2}}, y_{n+\frac{9}{2}}\right) \\ &+32K\left(x_{n+6}, x_{n+\frac{19}{4}}, y_{n+\frac{19}{4}}\right) \\ &+7K\left(x_{n+6}, x_{n+5}, y_{n+5}\right) \end{aligned}\right\} \\ & + \frac{h}{90}\left\{\begin{aligned} &7K\left(x_{n+6}, x_{n+5}, y_{n+5}\right) + 32K\left(x_{n+6}, x_{n+\frac{21}{4}}, y_{n+\frac{21}{4}}\right) \\ &+12K\left(x_{n+6}, x_{n+\frac{11}{2}}, y_{n+\frac{11}{2}}\right) + 32K\left(x_{n+6}, x_{n+\frac{23}{4}}, y_{n+\frac{23}{4}}\right) \\ &+7K\left(x_{n+6}, x_{n+6}, y_{n+6}\right) \end{aligned}\right\} \end{aligned} \tag{14}$$

The unknown values $y_{n+\frac{17}{4}}, y_{n+\frac{9}{2}}, y_{n+\frac{19}{4}}$ are found by using formula in (13). Lagrange interpolation at points $\{x_{n+2}, x_{n+3}, x_{n+4}, x_{n+5}, x_{n+6}\}$ is used to calculate for unknown values $y_{n+\frac{21}{4}}, y_{n+\frac{11}{2}}, y_{n+\frac{23}{4}}$. The following formulas have been derived:

$$\begin{aligned} y_{n+\frac{21}{4}} &= -\frac{45}{2048}y_{n+2} + \frac{65}{512}y_{n+3} - \frac{351}{1024}y_{n+4} \\ &\quad + \frac{585}{512}y_{n+5} + \frac{195}{2048}y_{n+6} \end{aligned}$$

$$y_{n+\frac{11}{2}} = -\frac{5}{128}y_{n+2} + \frac{7}{32}y_{n+3} - \frac{35}{64}y_{n+4} + \frac{35}{32}y_{n+5} + \frac{35}{128}y_{n+6} \quad (15)$$

$$y_{n+\frac{23}{4}} = -\frac{77}{2048}y_{n+2} + \frac{105}{512}y_{n+3} - \frac{495}{1024}y_{n+4} + \frac{385}{512}y_{n+5} + \frac{1155}{2048}y_{n+6}$$

$$z_{n+7} = \frac{2h}{45}\sum_{i=0}^{n+4}\omega_i^s K\left(x_{n+7},x_i,y_i\right) + \frac{h}{90}\begin{Bmatrix} 7K\left(x_{n+7},x_{n+4},y_{n+4}\right) \\ +32K\left(x_{n+7},x_{n+\frac{17}{4}},y_{n+\frac{17}{4}}\right) \\ +12K\left(x_{n+7},x_{n+\frac{9}{2}},y_{n+\frac{9}{2}}\right) \\ +32K\left(x_{n+7},x_{n+\frac{19}{4}},y_{n+\frac{19}{4}}\right) \\ +7K\left(x_{n+7},x_{n+5},y_{n+5}\right) \end{Bmatrix}$$

$$+\frac{h}{90}\begin{Bmatrix} 7K\left(x_{n+7},x_{n+5},y_{n+5}\right) + 32K\left(x_{n+7},x_{n+\frac{21}{4}},y_{n+\frac{21}{4}}\right) \\ +12K\left(x_{n+7},x_{n+\frac{11}{2}},y_{n+\frac{11}{2}}\right) + 32K\left(x_{n+7},x_{n+\frac{23}{4}},y_{n+\frac{23}{4}}\right) \\ +7K\left(x_{n+7},x_{n+6},y_{n+6}\right) \end{Bmatrix}$$

$$+\frac{h}{90}\begin{Bmatrix} 7K\left(x_{n+7},x_{n+6},y_{n+6}\right) + 32K\left(x_{n+7},x_{n+\frac{25}{4}},y_{n+\frac{25}{4}}\right) \\ +12K\left(x_{n+7},x_{n+\frac{13}{2}},y_{n+\frac{13}{2}}\right) + 32K\left(x_{n+7},x_{n+\frac{27}{4}},y_{n+\frac{27}{4}}\right) \\ +7K\left(x_{n+7},x_{n+7},y_{n+7}\right) \end{Bmatrix}. \quad (16)$$

The unknown values $y_{n+\frac{17}{4}}, y_{n+\frac{9}{2}}, y_{n+\frac{19}{4}}, y_{n+\frac{21}{4}}, y_{n+\frac{11}{2}}, y_{n+\frac{23}{4}}$ are found by using formula in (13) and (15). Lagrange interpolation at points $\{x_{n+3}, x_{n+4}, x_{n+5}, x_{n+6}, x_{n+7}\}$ is used to calculate for unknown values $y_{n+\frac{25}{4}}, y_{n+\frac{13}{2}}, y_{n+\frac{27}{4}}$. The following formulas are obtained:

$$y_{n+\frac{25}{4}} = -\frac{45}{2048}y_{n+3} + \frac{65}{512}y_{n+4} - \frac{351}{1024}y_{n+5} + \frac{585}{512}y_{n+6} + \frac{195}{2048}y_{n+7}$$

$$y_{n+\frac{13}{2}} = -\frac{5}{128}y_{n+3} + \frac{7}{32}y_{n+4} - \frac{35}{64}y_{n+5} + \frac{35}{32}y_{n+6} + \frac{35}{128}y_{n+7} \quad (17)$$

$$y_{n+\frac{27}{4}} = -\frac{77}{2048}y_{n+3} + \frac{105}{512}y_{n+4} - \frac{495}{1024}y_{n+5} + \frac{385}{512}y_{n+6} + \frac{1155}{2048}y_{n+7}$$

The stability of the proposed two point three-step block method together with the Boole's rule are investigated. The following linear test equation for the stability is given:

$$y'(x) = \xi y(x) + \eta\int_0^x y(s)\,ds \quad (18)$$

The solutions of (18) tend to zero as x → ∞ if and only if $\xi < 0$ and $\eta < 0$. Then, the region of absolute stability is the set of points $(h\xi, h^2\eta)$ for which all zeros of the stability polynomial,

$$\pi(r,h\xi,h^2\eta) := \tilde{\rho}(r)\left[\rho(r) - h\xi\sigma(r)\right] - h^2\eta\tilde{\sigma}(r)\sigma(r) \quad (19)$$

lie in the interior of the unit disk. From (19), the correspond unique polynomials ρ, σ, $\tilde{\rho}$ and $\tilde{\sigma}$ are given as

I. First point of corrector formula

$$\rho(r) = r^3 - r^2 \qquad \sigma(r) = -\frac{19}{720}r^4 + \frac{346}{720}r^3 + \frac{456}{720}r^2 - \frac{74}{720}r + \frac{11}{720} \quad (20)$$

II. Second point of corrector formula

$$\rho(r) = r^4 - r^2 \qquad \sigma(r) = \frac{29}{90}r^4 + \frac{124}{90}r^3 + \frac{24}{90}r^2 + \frac{4}{90}r - \frac{1}{90} \quad (21)$$

III. Boole's rule

$$\tilde{\rho}(r) = r^4 - 1 \qquad \tilde{\sigma}(r) = \frac{14}{45}r^4 + \frac{64}{45}r^3 + \frac{24}{45}r^2 + \frac{64}{45}r + \frac{14}{45} \quad (22)$$

Then, substitute (20), (21) and (22) into the formula of the stability polynomial as in (19). From the stability polynomial, the region of absolute stability of the combinations method is plotted. From Figure 2, the method is stable within the shaded region.

NUMERICAL RESULTS

We have tested five numerical problems that consist of linear and non-linear VIDEs and it involve $K(x, s) = 1$ and $K(x, s) \neq 1$. The results obtained were given in Tables 1 to

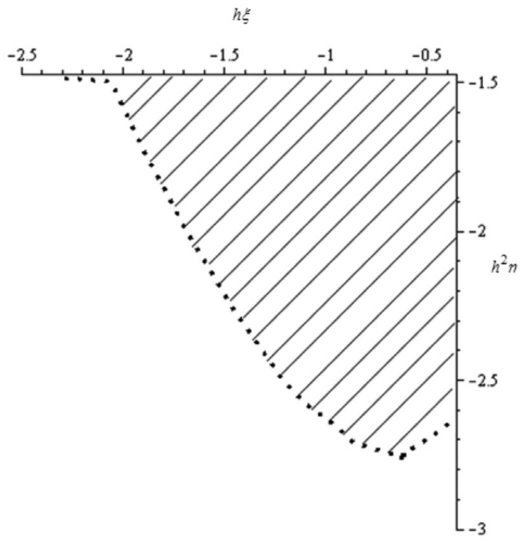

FIGURE 2. Stability region in the $h\xi$, $h^2\eta$ plane

5 in terms of maximum error, total steps, total function calls and timing. The notations used in the table are as follows:

MAXE	:	Maximum error
h	:	Step size
TS	:	Total steps
TFC	:	Total functions call
Time	:	Execution time in seconds
-	:	Not discuss by the author of the method
2P3BVIDE	:	Two point three-step block method as in this research
GBDF-5	:	Combination of boundary value methods and fifth order generalized backward differentiation formula by Chen and Zhang (2011)
ABM5	:	Fifth order Adams-Bashforth-Moulton predictor-corrector method in Faires and Burden (2005)

Problem 1 ($K(x, s) = 1$) Linear VIDEs:

$$y'(x) = -\int_0^x y(s)\,ds \qquad y(0) = 1 \quad 0 \le x \le 1$$

Exact solution: $y(x) = \cos x$.

Problem 2 ($K(x, s) = 1$) Linear VIDEs:

$$y'(x) = 1 - \int_0^x y(s)\,ds \qquad y(0) = 0 \quad 0 \le x \le 1$$

Exact solution: $y(x) = \sin x$.

Problem 3 ($K(x, s) \neq 1$) Linear VIDEs:

$$y'(x) = -\sin x - \cos x + \int_0^x 2\cos(x-s)\,y(s)\,ds$$

$$y(0) = 1 \qquad 0 \le x \le 5$$

Exact solution: $y(x) = e^{-x}$.

Problem 4 ($K(x, s) \neq 1$) Nonlinear VIDEs:

$$y'(x) = xe^{\left(1-y(x)\right)} - \frac{1}{\left(1+x\right)^2} - x - \int_0^x \frac{x}{\left(1+s\right)^2} e^{\left(1-y(s)\right)}\,ds$$

$$y(0) = 1 \qquad 0 \le x \le 4$$

Exact solution: $y(x) = \frac{1}{1+x}$.

Problem 5 ($K(x, s) \neq 1$) Nonlinear VIDEs:

$$y'(x) = 2x - \frac{1}{2}\sin\left(x^4\right) + \int_0^x x^2 s \cos\left(x^2 y\left(s\right)\right) ds$$

$$y(0) = 0 \qquad 0 \le x \le 2$$

Exact solution: $y(x) = x^2$.

DISCUSSION AND CONCLUSION

In this section, the performance of the proposed multistep block method with quadrature rule in terms accuracy, total function calls and execution times for solving the five numerical problems is presented. It is important to mention that the comparison is being made with ABM5 which has been run in the same environment as the 2P3BVIDE.

Tables 1 and 2 display the numerical results for the linear VIDEs problem when $K(x, s) = 1$ and it shown that the maximum error of the 2P3BVIDE is one or two decimal places better in terms of accuracy compared to ABM5. Table 3 represents the results for the linear VIDEs when $K(x, s) \neq 1$ and we could observe that the GBDF-5 outperformed the 2P3BVIDE by obtaining smaller maximum error at smaller h but the 2P3BVIDE manage to give more accurate approximation at larger step sizes. The accuracies are comparable between ABM5 and 2P3BVIDE. In terms of total steps, total function calls and timing, we could observed that the 2P3BVIDE is less costly compared to ABM5.

The nonlinear problems of VIDEs when $K(x, s) \neq 1$ are solved and the numerical results are shown in Tables 4 and 5. We could observe that the maximum error is comparable between ABM5 and 2P3BVIDE. The results also showed that the 2P3BVIDE manage to obtain less total number of steps and function call compared to ABM5. The proposed 2P3BVIDE was represented in a block manner and it is able to approximate the solutions at two points simultaneously. Therefore, the proposed multistep block method managed to achieve the execution time faster than the existing method and yet manage to produce better accuracy.

TABLE 1. Numerical results for Problem 1

h	Method	MAXE	TS	TFC	Time
$\frac{1}{40}$	ABM5	2.8951e-007	40	88	0.0940
	2P3BVIDE	5.7323e-008	22	50	0.0574
$\frac{1}{80}$	ABM5	3.6127e-008	80	168	0.1783
	2P3BVIDE	3.5893e-009	42	90	0.1254
$\frac{1}{160}$	ABM5	4.3953e-009	160	328	0.2370
	2P3BVIDE	2.2443e-010	82	170	0.1926
$\frac{1}{320}$	ABM5	5.4213e-010	320	648	0.3525
	2P3BVIDE	1.3908e-011	162	330	0.3277
$\frac{1}{640}$	ABM5	6.7325e-011	640	1288	0.5971
	2P3BVIDE	8.6930e-013	640	650	0.5000
$\frac{1}{1280}$	ABM5	8.3668e-012	1280	2568	1.0786
	2P3BVIDE	5.4179e-014	642	1290	1.0293

TABLE 2. Numerical results for Problem 2

h	Method	MAXE	TS	TFC	Time
$\frac{1}{40}$	ABM5	4.4529e-009	40	88	0.1020
	2P3BVIDE	1.2349e-009	22	50	0.0700
$\frac{1}{80}$	ABM5	2.3862e-010	80	168	0.1579
	2P3BVIDE	3.8642e-011	42	90	0.1166
$\frac{1}{160}$	ABM5	1.4271e-011	160	328	0.2622
	2P3BVIDE	1.2080e-012	82	170	0.2034
$\frac{1}{320}$	ABM5	8.6009e-013	320	648	0.4222
	2P3BVIDE	3.7751e-014	162	330	0.3124
$\frac{1}{640}$	ABM5	4.7296e-014	640	1288	0.6262
	2P3BVIDE	5.3291e-015	322	650	0.5000
$\frac{1}{1280}$	ABM5	1.6764e-014	1280	2568	1.1360
	2P3BVIDE	1.3545e-014	642	1290	0.9598

TABLE 3. Numerical results for Problem 3

h	Method	MAXE	TS	TFC	Time
$\frac{1}{4}$	GBDF-5	2.3922e-002	-	-	-
	ABM5	8.1337e-003	20	85	0.0715
	2P3BVIDE	6.1138e-003	11	59	0.0462
$\frac{1}{8}$	GBDF-5	3.1790e-004	0	-	-
	ABM5	4.7616e-004	40	165	0.1623
	2P3BVIDE	3.9009e-004	21	99	0.0900
$\frac{1}{16}$	GBDF-5	4.3708e-006	0	-	-
	ABM5	2.1034e-005	80	325	0.2139
	2P3BVIDE	1.6881e-005	41	179	0.1930
$\frac{1}{32}$	GBDF-5	7.5567e-008	-	-	-
	ABM5	7.8509e-007	160	645	0.3278
	2P3BVIDE	6.1208e-007	81	339	0.2494
$\frac{1}{64}$	GBDF-5	-	-	-	-
	ABM5	2.6828e-008	320	1285	0.5850
	2P3BVIDE	2.0516e-008	161	659	0.4239
$\frac{1}{128}$	GBDF-5	-	-	-	-
	ABM5	8.7684e-010	640	2565	1.1659
	2P3BVIDE	6.6334e-010	321	1299	0.7499

TABLE 4. Numerical results for Problem 4

h	Method	MAXE	TS	TFC	Time
$\frac{1}{40}$	ABM5	1.7212e-008	160	645	0.3430
	2P3BVIDE	8.3237e-008	81	339	0.2850
$\frac{1}{80}$	ABM5	3.0551e-009	320	1285	0.5544
	2P3BVIDE	3.8384e-009	161	659	0.3879
$\frac{1}{160}$	ABM5	1.9089e-010	640	2565	1.1720
	2P3BVIDE	2.0775e-010	321	1299	0.6778
$\frac{1}{320}$	ABM5	1.1926e-011	1280	5125	1.8678
	2P3BVIDE	1.2654e-011	641	2579	1.4850
$\frac{1}{640}$	ABM5	7.4529e-013	2560	10245	3.9401
	2P3BVIDE	9.6889e-013	1281	5139	2.5689
$\frac{1}{1280}$	ABM5	4.6518e-014	5120	20485	8.4150
	2P3BVIDE	4.3676e-013	2561	10259	6.2365

TABLE 5. Numerical results for Problem 5

h	Method	MAXE	TS	TFC	Time
$\frac{2}{9}$	ABM5	7.2747e-002	7	41	0.0520
	2P3BVIDE	6.8284e-002	5	35	0.0470
$\frac{2}{17}$	ABM5	7.8868e-003	15	73	0.0730
	2P3BVIDE	8.4729e-003	9	51	0.0680
$\frac{2}{33}$	ABM5	8.9015e-005	31	137	0.1355
	2P3BVIDE	9.3109e-005	17	83	0.0780
$\frac{2}{65}$	ABM5	2.9296e-007	63	265	0.2133
	2P3BVIDE	3.0567e-007	33	147	0.1560
$\frac{2}{129}$	ABM5	7.1445e-009	127	521	0.2919
	2P3BVIDE	7.0325e-009	65	275	0.2501
$\frac{2}{257}$	ABM5	1.3086e-010	255	1033	0.4532
	2P3BVIDE	1.2241e-010	129	531	0.3430

In conclusion, the proposed multistep block method based on the two point three-step block method with the quadrature Boole's rule is suitable for solving the second kind VIDEs.

ACKNOWLEDGEMENTS

This work was supported by the Fundamental Research Grant Scheme (FRGS: 5524973) from the Ministry of Education, Malaysia and Graduate Research Fund (GRF) from Universiti Putra Malaysia.

REFERENCES

Chang, S.H. 1982. On certain extrapolation methods for the numerical solution of integro-differential equations. *Mathematics of Computation* 39(159): 165-171.

Chen, H. & Zhang, C. 2011. Boundary value methods for Volterra integral and integro-differential equations. *Applied Mathematics and Computation* 218: 2619-2630.

Day, J.T. 1967. Note on the numerical solution of integro-differential equations. *The Computer Journal* 9(4): 394-395.

Dehghan, M. & Salehi, R. 2012. The numerical solution of the non-linear integro-differential equations based on the meshless method. *Journal of Computational and Applied Mathematics* 236: 2367- 2377.

Faires, D. & Burden, R.L. 2005. *Numerical Analysis*. Belmont, CA: Thomson Brooks/Cole.

Feldstein, A. & Sopka, J.R. 1974. Numerical methods for nonlinear Volterra integro-differential equations. *Siam J. Numer. Anal*. 11: 826-846.

Filiz, A. 2014. Numerical solution of linear Volterra integro-differential equation using Runge-Kutta-Fehlberg method. *Applied and Computational Mathematics* 3(1): 9-14.

Filiz, A. 2013. A fourth-order robust numerical method for integro-differential equations. *Asian Journal of Fuzzy and Applied Mathematics* 1(1): 28-33.

Ishak, F. & Ahmad, S.N. 2016. Development of extended trapezoidal method for numerical solution of Volterra integro-differential equations. *International Journal of Mathematics, Computational, Physical, Electrical and Computer Engineering* 10(11): 52856.

Kürkçü, Ö.K., Aslan, E. & Sezer, M. 2017. A novel collocation method based on residual error analysis for solving integro-differential equations using hybrid Dickson and Taylor polynomials. *Sains Malaysiana* 46(2): 335-347.

Lambert, L.D. 1973. *Computational Methods in Ordinary Differential Equations*. New York: John Wiley & Sons, Inc.
Linz, P. 1969. Linear multistep methods for Volterra integro-differential equations. *Journal of the Association for Computing Machinery* 16(2): 295-301.
Lubich, C. 1982. Runge-Kutta theory for Volterra integro-differential equations. *Numer. Math* 40: 119-135.
Majid, Z.A. & Suleiman, M. 2011. Predictor-corrector block iteration method for solving ordinary differential equations. *Sains Malaysiana* 40(6): 659-664.
Makroglou, A. 1982. Hybrid methods in the numerical solution of Volterra integro-differential equations. *IMA Jounal of Numerical Analysis* 2: 21-35.
Mocarsky, W.L. 1971. Convergence of step-by-step methods for non-linear integro-differential equations. *IMA Journal of Applied Mathematics* 8(2): 235-239.
Mohamed, N.A. & Majid, Z.A. 2016. Multistep block method for solving Volterra integro-differential equations. *Malaysian Journal of Mathematical Sciences* 10: 33-48.
Yuan, W. & Tang, T. 1990. The numerical analysis of implicit Runge-Kutta methods for a certain nonlinear integro-differential equation. *Mathematics of Computation* 54(189): 155-168.

Zanariah Abdul Majid* & Nurul Atikah Mohamed
Institute for Mathematical Research
Universiti Putra Malaysia
43400 UPM Serdang, Selangor Darul Ehsan
Malaysia

Zanariah Abdul Majid*
Mathematics Department, Faculty of Science
Universiti Putra Malaysia
43400 UPM Serdang, Selangor Darul Ehsan
Malaysia

*Corresponding author; email: zana_majid99@yahoo.com

Received: 3 July 2018
Accepted: 21 November 2018

.