Sains Malaysiana 52(8)(2023): 2337-2351
*http://doi.org/10.17576/jsm-2023-5208-13*

# Hybrid Multistep Block Method for Solving Neutral Volterra Integro-Differential Equation with Proportional and Mixed Delays

(Kaedah Berbilang Langkah Blok Hibrid untuk Menyelesaikan Persamaan Kamiran-Pembezaan Neutral Volterra dengan Kelengahan Berkadar dan Bercampur)

NUR INSHIRAH NAQIAH ISMAIL[1] & ZANARIAH ABDUL MAJID[1,2,*]

[1]*Institute for Mathematical Research, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*
[2]*Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*

*Received: 20 April 2023/Accepted: 11 July 2023*

## ABSTRACT

The neutral Volterra integro-differential equation with proportional and mixed delays (NDVIDE) is being solved by a newly proposed technique in numerical method, namely, the two-point one off-point block multistep method (1OBM3). The method is also known as a hybrid multistep block method. Subsequently, Lagrange interpolating polynomial is utilized in order to develop the hybrid block method. The foundation of the technique is taken from predictor and corrector formulae. The proposed method will solve NDVIDE in two steps simultaneously, with three predictor formulae including one off-point. The NDVIDE problems are solved via the constant step size technique. In order to solve the integral and differential parts of the problems, two alternative numerical approaches are applied. The differentiation part is approximated by deriving the divided difference formula, while the integration part is interpolated using composite Simpson's rule. Note that the proposed method has been analysed thoroughly regarding its order, consistency, zero stability and convergence of the method. The stability region for 1OBM3 has been constructed based on the stability polynomial obtained. Consequently, numerical results are presented to demonstrate the effectiveness of the proposed method, 1OBM3.

Keywords: Hybrid multistep block method; mixed delay; neutral delay Volterra integro-differential equations; proportional delay

## ABSTRAK

Persamaan kamiran-pembezaan neutral Volterra dengan kelengahan berkadar dan bercampur (NDVIDE) diselesaikan dengan teknik baharu yang dicadangkan dalam kaedah berangka iaitu, kaedah blok berbilang langkah dua titik dan satu luar-titik (1OBM3). Kaedah ini juga dikenali sebagai kaedah blok berbilang langkah hibrid. Interpolasi polinomial Lagrange dimanfaatkan bagi membangunkan kaedah blok hibrid. Asas kepada kaedah ini diambil daripada formula peramal-pembetul. Kaedah yang dicadangkan akan menyelesaikan NDVIDE dalam dua langkah serentak dengan tiga formula peramal termasuk satu luar-titik. Masalah NDVIDE diselesaikan melalui teknik saiz langkah malar. Untuk menyelesaikan masalah di bahagian kamiran dan pembezaan, dua pendekatan berangka alternatif digunakan. Bahagian pembezaan dianggarkan dengan memperoleh formula perbezaan terbahagi manakala bahagian kamiran di interpolasi dengan menggunakan peraturan Simpson komposit. Kaedah yang dicadangkan telah dianalisis dengan teliti dari segi peringkat, ketekalan, kestabilan sifar dan penumpuan. Kawasan kestabilan untuk 1OBM3 telah dibina berdasarkan polinomial kestabilan yang diperoleh. Keputusan berangka dibentangkan untuk menunjukkan keberkesanan kaedah 1OBM3 yang dicadangkan.

Kata kunci: Kaedah blok berbilang langkah hybrid; kelengahan bercampur; kelengahan berkadar; kelengahan neutral persamaan kamiran-pembezaan Volterra

## INTRODUCTION

Vito Volterra introduced the Volterra integral and integro-differential equations in 1926 (Altun (2021a). These equations have been implemented widely in technical areas, especially science, technology, and engineering. Delay differential equations/systems and Volterra integro-differential equations (VIDEs), which are well-known mathematical models in the related literature, have been used in various real-world applications, including electrical circuits, the process of making glass, biology, physics, chemistry, control theory, and economics. The proportional delay (the pantograph equation) is a time delay system. However, unlike other time delay systems, it operates proportionally. Time delays typically appear in sensor and actuator-based feedback loops. They are constantly presented in a structural testing method called real-time dynamic substructuring. When an output of a system is returned to or becomes the input in the following cycle that creates a circuit or loop, a feedback loop is created. Note that these transmission systems are frequently used in communication technologies. The transmission method for approximated indicators to a remote-control centre is becoming simpler due to the quick development of communication technologies, providing more opportunities for researchers to propose other solutions to these issues (Ismail, Majid & Senu 2020). Integral with time lags are considered to illustrate the model realistically, given as follows,

$$\begin{aligned} y'(x) &= f\big(x, y(x), z(x)\big), \quad x \in [a,b] \\ y(x) &= \phi(x), \end{aligned} \tag{1}$$

where

$$z(x) = \int_x^{\alpha} K\left(t, y(t), y(\alpha), y'(\alpha)\right) dt.$$

$\alpha$ is a mixed delay since it could involve any kind of delay, including proportional, constant, time-dependent, and state-dependent delays. The specified starting function, $y(x) = \phi(x)$, is the initial value provided wherever $\tau = 0$, which results in Equation (1) to be reduced to a standard initial value problem (IVP). Note that the expression of $y(\alpha)$ and $y'(\alpha)$ are the delay solutions where $\alpha$ is the delay argument. The proportional delay also plays an important role in industry and is known as the pantograph equation where the general form is modelled as follows:

$$\begin{aligned} y'(x) &= f\big(x, y(x), z(x)\big), \quad x \in [a,b] \\ y(x) &= \phi(x), \end{aligned} \tag{2}$$

where

$$z(x) = \int_x^{qx} K\left(t, y(t), y(qx), y'(qx)\right) dt.$$

Here, $0 < q < 1$ is the restricted ratio for the proportional delay, $qx$ is the delay term while $y(qx)$ and $y'(qx)$ represent the delay solutions. The proposed method, 1OBM3, will be derived to solve Equations (1) and (2). Composite quadrature rule (Newton-Cote) is the integration formula required to solve an integral part implicitly.

According to the previous literature, an efficient numerical method to solve NDVIDE with constant type and retarded delay Volterra integro-differential equation (RDVIDE) with pantograph equation has been presented by Mirzaee, Bimesl and Tohidi (2016) using both operational matrices of differentiation and delay based on Euler polynomials. Since the issues are frequently challenging to resolve analytically, a numerical approach is necessary. The error estimation of the method is also provided whereas if N is sufficiently large enough, the errors decrease. Obviously, the present method can be easily extended and applied to multi-dimensional integro-differential equations. A while later, a new backward substitution method for linear functional arguments based on VIDEs with neutral delay multipoint boundary value problems was proposed by Reutskiy (2016). In the suggested method, the initial equation is substituted for an approximate equation with a set of free parameters and an exact analytic solution. In this study, only polynomial functions are applied. However, the framework of the suggested method also allows for the utilization of trigonometric and radial basis functions.

Later, the research from Wen and Yu (2016) examined the convergence of numerical techniques for NDVIDEs' IVPs. For a class of nonlinear systems of NDVIDE, the Runge-Kutta methods' error estimation is derived. The theoretical results are supported by a few numerical experiments that were provided. Note that the research could still be extended to the one-leg, multistep, Runge-Kutta, and any general linear methods. Consequently, Wen and Zhou (2017) have provided the error analysis of one-leg methods for a class of nonlinear NDVIDE proving that an A-stable one-leg method with an appropriate quadrature rule applied to NDIDEs is convergent. Numerical results further support the theoretical findings.

A numerical method is proposed by Yuzbasi and Karacayir (2017) to solve high-order linear Volterra delay integro-differential equations. In the studies, a power series can represent the exact solution. By calculating the inaccuracy, the method seeks to improve the accuracy of the approximate solutions. Vijayakumar (2018) considered a class of abstract neutral integro-differential inclusions with infinite delay in Hilbert spaces. The author solved the problems by establishing Bohnenblust–

Karlin's fixed point theorem. Correspondingly, the global exponential stability (GES) of the zero solution of a nonlinear NDVIDE with variable lags has been investigated by Altun (2021b). A new stability criterion is derived based on the Lyapunov functional approach. It has been noted that the simulation results validate the efficacy and precision of the study's theoretical findings. Moreover, Altun (2021b) reconsidered the asymptotic behaviours of solutions to the NDVIDE problem where he obtained novel sufficient conditions to establish it using the Lyapunov method. Subsequently, a particular case for NDVIDE is solved via the differential transformation method (DTM).

In the same year, an effective numerical technique was introduced by Gurbuz (2021) for finding the solutions to first-order integro-differential equations, including neutral terms with mixed delays. The delays include constant, time-dependent (variable delay), and state-dependent delays. Consequently, an alternative numerical method is expressed by fundamental matrices, Laguerre polynomials with matrix forms. Prospects include the extension of this numerical analysis and technique to further models that involve Volterra integro-differential equations, including those with retarded delay terms. However, some adjustments are necessary. More recently, the Taylor collocation method has been applied by Laib, Bellour and Boulmerka (2022) to numerically solve a $k^{th}$-order linear NDVIDE with constant delay and variable coefficients. The method is convergent with good accuracy and easy to implement. Further research on this kind of problem will be conducted by generalizing the work done to a system of $k^{th}$-order NDVIDE.

A detailed examination of the literature uncovers several gaps in solving NDVIDE. Hence, this study aims to create a two-step one-point (hybrid) multistep block method to solve NDVIDE of mixed and proportional delay types. This has not been done by any researcher in the antecedent literature. Therefore, the formulation of the suggested method for dealing with the problems is the main purpose of this research.

## DERIVATION OF HYBRID MULTISTEP BLOCK METHOD

In agreement with Lambert (1991), a hybrid method is an extension of a multistep method which involves off-point in the general form of linear multistep method (LMM). The formulation of third order two-point one off-step multistep block method (1OBM3) is based on a Lagrange polynomial shown below,

$$\begin{aligned} P(x) &= L_{n,0}(x)f(x_0) + \cdots + L_{n,n}(x)f(x_n) \\ &= \sum_{k=0}^{n} f(x_k)L_{n,k}(x) \end{aligned} \tag{3}$$

where

$$L_{n,k}(x) = \prod_{\substack{i=0 \\ i\neq k}}^{n} \frac{(x-x_i)}{(x_k-x_i)},$$

$$k = 0,1,\ldots,n,$$

where $p$ denotes the order of the method. The linear difference operator L is associated with $k$-step hybrid formula based on:

$$\begin{aligned} L[y(x){:}\,h] = \sum_{j=0}^{k}\left[\alpha_j y(x+jh) - h\beta_j y'(x+jh)\right] \\ - h\beta_v y'(x+vh). \end{aligned} \tag{4}$$

The evaluation of both points, $y_1\,(x_{n+1})$ and $y_2\,(x_{n+2})$ as well as their corresponding delay and delay derivative solutions are shown in this subsection to derive 1OBM3 where $\alpha_k = 1$, $\alpha_0$ and $\beta_0$ are not equals to zero, while $v \notin \{0,1, \ldots, k\}$, as stated in Lambert (1973). In this research, the off-step method is depicted in Figure 1:

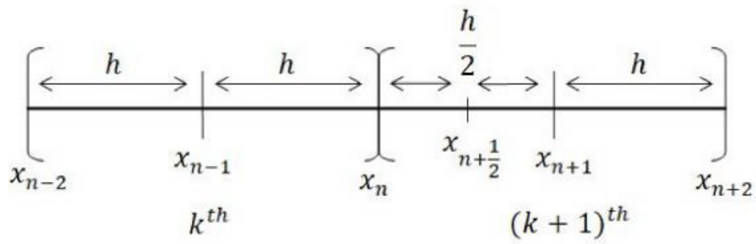


FIGURE 1. Hybrid multistep block method

Supposed that the first $k^{th}$ block compromised $x_{n-2}$, $x_{n-1}$ and $x_n$ where $x_{n-2}$ represented the starting point whereas $x_n$ represent the final point in the current frame, as shown in Figure 1. The starting values for $(k+1)^{th}$ block is established by estimations from $k^{th}$ block. Before evaluating $y_{n+1}$ and $y_{n+2}$, the off-point iteration, $y_{n+\frac{1}{2}}$, will be computed. The subsequent block will be calculated according to the identical process until the interval's endpoint. Subsequently, the approach tries synchronously to resolve the problem. As a two-point block method, the suggested technique will produce two solutions in a single block. The approximate solutions for $y_{n+1}$ and $y_{n+2}$ will be developed by applying Lagrange interpolating polynomial, Equation (3). Hence, the first-order general linear hybrid multistep method for the first point corrector of 1OBM3 is shown as follows:

$$\int_{x_n}^{x_{n+1}} y'(x)dx = \int_{x_n}^{x_{n+1}} [f(x, y(x), z(x))]dx,$$

$$y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} [f(x, y(x), z(x))]dx,$$

by replacing the $[f(x, y(x), z(x))]$ with $P_{1,2}$:

$$\begin{aligned} y(x_{n+1}) - y(x_n) &= \int_{x_n}^{x_{n+1}} P_{1,2}\, dx, \\ y(x_{n+1}) &= y(x_n) + \int_{x_n}^{x_{n+1}} P_{1,2}\, dx, \end{aligned} \tag{5}$$

where $P_{1,2}$ is the second-degree Lagrange interpolating polynomial that will be applied to derive the function $f(x, y(x), z(x))$ with $h = x_{n+1} - x_n$. The derived method of order three requires applying a second-degree Lagrange interpolating polynomial with three points needed for all predictors and correctors formulae. The points needed for the first point corrector formula are listed as follows:

$$\{x_{n+1}, f_{n+1}\}, \left\{x_{n+\frac{1}{2}}, f_{n+\frac{1}{2}}\right\}, \{x_{n-1}, f_{n-1}\}.$$

The implementation of Lagrange with second-degree is shown herewith:

$$\begin{aligned} P_{1,2}^c(x) = &\frac{(x - x_{n+\frac{1}{2}})(x - x_{n-1})}{(x_{n+1} - x_{n+\frac{1}{2}})(x_{n+1} - x_{n-1})} f_{n+1} \\ &+ \frac{(x - x_{n+1})(x - x_{n-1})}{\left(x_{n+\frac{1}{2}} - x_{n+1}\right)\left(x_{n+\frac{1}{2}} - x_{n-1}\right)} f_{n+\frac{1}{2}} \\ &+ \frac{(x - x_{n+1})(x - x_{n+\frac{1}{2}})}{(x_{n-1} - x_{n+1})(x_{n-1} - x_{n+\frac{1}{2}})} f_{n-1}, \end{aligned} \tag{6}$$

where $P_{1,2}^c(x)$ represents the second-degree polynomial at the first point corrector formula. Whilst the second point corrector of 1OBM3 is shown as follows:

$$\int_{x_n}^{x_{n+2}} y'(x)dx = \int_{x_n}^{x_{n+2}} [f(x, y(x), z(x))]dx,$$

where the $[f(x, y(x), z(x))]$ with $P_{2,2}$

$$\begin{aligned} y(x_{n+2}) - y(x_n) &= \int_{x_n}^{x_{n+2}} P_{2,2}\, dx, \\ y(x_{n+2}) &= y(x_n) + \int_{x_n}^{x_{n+2}} P_{2,2}.\, dx, \end{aligned} \tag{7}$$

where the points required for the second point corrector formula are:

$$\{x_{n+2}, f_{n+2}\}, \left\{x_{n+\frac{1}{2}}, f_{n+\frac{1}{2}}\right\}, \{x_n, f_n\}.$$

The second-degree polynomial at second point corrector formula, $P_{2,2}^c(x)$, is given by:

$$\begin{aligned} P_{2,2}^c(x) = &\frac{(x - x_{n+\frac{1}{2}})(x - x_n)}{(x_{n+2} - x_{n+\frac{1}{2}})(x_{n+2} - x_n)} f_{n+2} \\ &+ \frac{(x - x_{n+2})(x - x_n)}{\left(x_{n+\frac{1}{2}} - x_{n+2}\right)\left(x_{n+\frac{1}{2}} - x_n\right)} f_{n+\frac{1}{2}} \\ &+ \frac{(x - x_{n+2})(x - x_{n+\frac{1}{2}})}{(x_n - x_{n+2})(x_n - x_{n+\frac{1}{2}})} f_n. \end{aligned} \tag{8}$$

Hence, taking $s = \frac{x - x_{n+2}}{h}$ in Equations (6) and (8) yields:

$$\begin{aligned} P_{1,2}^c(x) = &\frac{(\frac{3}{2}h + sh)(3h + sh)}{(\frac{1}{2}h)(2h)} f_{n+1} + \frac{(h + sh)(3h + sh)}{\left(-\frac{1}{2}h\right)\left(\frac{3}{2}h\right)} f_{n+\frac{1}{2}} \\ &+ \frac{(h + sh)(\frac{3}{2}h + sh)}{(-2h)(-\frac{3}{2}h)} f_{n-1}, \end{aligned}$$

$$\begin{aligned} P_{2,2}^c(x) = &\frac{(\frac{3}{2}h + sh)(2h + sh)}{(\frac{3}{2}h)(2h)} f_{n+2} + \frac{(h + sh)(2h + sh)}{\left(-\frac{3}{2}h\right)\left(\frac{1}{2}h\right)} f_{n+\frac{1}{2}} \\ &+ \frac{(sh)(\frac{3}{2}h + sh)}{(-2h)(-\frac{1}{2}h)} f_n. \end{aligned}$$

Then, we replace $dx = h\, ds$ and solve Equations (5) and (7) using MAPLE software. The predictor formulae are obtained following the same procedures as the corrector formulae. Hence, the developed 1OBM3 is obtained as shown herewith:

$$
\begin{aligned}
y^p_{n+\frac{1}{2}} &= y_n + \frac{h}{24}(17f_n - 7f_{n-1} + 2f_{n-2}),\\
y^p_{n+1} &= y_n + \frac{h}{12}(23f_n - 16f_{n-1} + 5f_{n-2}),\\
y^p_{n+2} &= y_n + \frac{h}{3}(19f_n - 20f_{n-1} + 7f_{n-2}),\\
y^c_{n+1} &= y_n + \frac{h}{36}\left(3f_{n+1} + 32f_{n+\frac{1}{2}} + f_{n-1}\right),\\
y^c_{n+2} &= y_n + \frac{h}{9}\left(5f_{n+2} + 16f_{n+\frac{1}{2}} - 3f_n\right).
\end{aligned} \tag{9}
$$

The derived method, 1OBM3 will be applied in solving NDVIDE with mixed and proportional delays.

## ORDER, ERROR CONSTANT AND LOCAL TRUNCATION ERROR OF HYBRID MULTISTEP BLOCK METHOD

According to Jator (2010), Definition 1 obtains the suggested order and error constant for 1OBM3. Note that, the order of a numerical method quantifies the reduction in error of a numerical solution with decreasing step size.

*Definition 1* The proposed off-step block method, Equation (9), is of order $s$ if, $C_0 = C_1 = \cdots = C_s = 0$ while its error constant is $C_{s+1} \neq 0$ where $s = 2,3,\ldots$,

$$C_0 = \sum_{j=0}^{k} \alpha_j,$$

$$C_1 = \sum_{j=0}^{k} j\,\alpha_j - \sum_{j=0}^{k} \beta_j - \sum_{j=1}^{1} \beta_{v_j},$$

$$\vdots$$

$$C_s = \frac{1}{s!}\left[\sum_{j=1}^{k} j^s\,\alpha_j - s\left[\sum_{j=1}^{k} j^{(s-1)}\,\beta_j + \sum_{j=1}^{1} v_j^{(s-1)}\,\beta_{v_j}\right]\right].$$

Based on Li and Li (2021), the definitions or theorems applied to ordinary differential equations (ODEs) could also be utilized in integro-differential problems. Hence, by letting $k = 3$, the corrector formulae can be rewritten as shown in the matrix form herewith:

$$
\begin{bmatrix} 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix}
\begin{bmatrix} y_{n-2} \\ y_{n-1} \\ y_n \\ y_{n+1} \\ y_{n+2} \end{bmatrix}
= h \begin{bmatrix} 0 & \frac{1}{36} & 0 & \frac{3}{36} & 0 \\ 0 & 0 & -\frac{3}{9} & 0 & \frac{5}{9} \end{bmatrix}
\begin{bmatrix} f_{n-2} \\ f_{n-1} \\ f_n \\ f_{n+1} \\ f_{n+2} \end{bmatrix}
+ h \begin{bmatrix} \frac{32}{36} \\ \frac{16}{9} \end{bmatrix} f_{n+\frac{1}{2}}, \tag{10}
$$

where

$$\alpha_0 = \begin{bmatrix}0\\0\end{bmatrix},\quad \alpha_1 = \begin{bmatrix}0\\0\end{bmatrix},\quad \alpha_2 = \begin{bmatrix}-1\\-1\end{bmatrix},\quad \alpha_3 = \begin{bmatrix}1\\0\end{bmatrix},\quad \alpha_4 = \begin{bmatrix}0\\1\end{bmatrix},$$

$$\beta_0 = \begin{bmatrix}0\\0\end{bmatrix},\quad \beta_1 = \begin{bmatrix}\frac{1}{36}\\0\end{bmatrix},\quad \beta_2 = \begin{bmatrix}0\\-\frac{3}{9}\end{bmatrix},\quad \beta_3 = \begin{bmatrix}\frac{3}{36}\\0\end{bmatrix},\quad \beta_4 = \begin{bmatrix}0\\\frac{5}{9}\end{bmatrix},$$

$$\beta_{v_1} = \begin{bmatrix}\frac{32}{36}\\\frac{16}{9}\end{bmatrix}.$$

As stated by Jator (2010), the hybrid method 1OBM3 is of order $s$ when following Definition 1 where,

$$C_0 = \sum_{j=0}^{4} \alpha_j = \begin{bmatrix}0\\0\end{bmatrix}$$

$$\vdots$$

$$C_3 = \frac{1}{3!}\left[\sum_{j=1}^{4} j^3\alpha_j - 3\left(\sum_{j=1}^{4} j^2\,\beta_j + \sum_{j=1}^{1} v_j^2\,\beta_{v_j}\right)\right] = \begin{bmatrix}0\\0\end{bmatrix}$$

$$C_4 = \frac{1}{4!}\left[\sum_{j=1}^{4} j^4\alpha_j - 4\left(\sum_{j=1}^{4} j^3\,\beta_j + \sum_{j=1}^{1} v_j^3\,\beta_{v_j}\right)\right] = \begin{bmatrix}\frac{1}{79}\\-\frac{1}{9}\end{bmatrix}.$$

Based on these evaluation, as $C_s = C_0 = C_1 = C_2 = C_3 = 0$, thus 1OBM3 is of order three with an error constant, $C_{s+1} = C_4 = \begin{bmatrix}\frac{1}{79}\\-\frac{1}{9}\end{bmatrix}$ where $C_{s+1} \neq 0$. Note that the local truncation error (LTE) is determined by referring to Baharum, Majid and Senu (2018).

*Definition 2* The local truncation error (LTE) at $x_{n+k}$ of the method is defined as expression $L[y(x_n); h]$, when $y(x)$ is the theoretical solution of the IVP given in, Equation (4). For the corrector formula, $y^c_{n+1}$, stated in Equation (9), the Taylor expansion will be applied to find the LTE, where,

$$y_{n+1} = y_n + hy'_n + h^2\frac{1}{2!}y''_n + h^3\frac{1}{3!}y'''_n + O(h^4),$$

while for the corrector formula, $y^c_{n+2}$, is given by,

$$y_{n+2} = y_n + 2hy'_n + 2h^2\frac{1}{2!}y''_n + 2h^3\frac{1}{3!}y'''_n + O(h^4).$$

Hence, the LTE for the proposed corrector method is $O(h^4)$.

## CONSISTENCY AND ZERO-STABILITY OF THE HYBRID MULTISTEP BLOCK METHOD

Consistency and zero-stability are essential characteristics in numerical analysis to ensure a numerical approach converges. Consistency is the ability of the numerical method to get progressively closer to solving the issue precisely as the step size or mesh size approaches zero. A numerical method is considered consistent if the step size approaches zero, and the truncation error also approaches zero. Referring to Lambert (1973), if both conditions listed in Equation are met, the multistep procedure is said to be consistent.

*Definition 3* The numerical method is said to be consistent if the order of method is $s \geq 1$ and the method is consistent if and only if,

$$(i) \sum_{j=0}^{k} \alpha_j = 0, \qquad (ii) \sum_{j=0}^{k} j\alpha_j = \sum_{j=0}^{k} \beta_j \quad (11)$$

Condition (*i*) has already been proven previously. Hence, condition (*ii*) is satisfied when,

$$\sum_{j=0}^{k} j\,\alpha_j = \sum_{j=0}^{4} j\,\alpha_j = \begin{bmatrix} 1 \\ 2 \end{bmatrix},$$

$$\sum_{j=0}^{k} \beta_j + \sum_{j=1}^{2} \beta_{v_j} = \sum_{j=0}^{4} \beta_j + \sum_{j=1}^{2} \beta_{v_j} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

Therefore, the method is consistent.

On the other hand, zero-stability refers to the property that the numerical method does not amplify errors over time. Specifically, a numerical method is said to be zero-stable if the errors introduced at each time step do not grow over time, and instead decay to zero as the number of time steps goes to infinity. In accordance with Baharum, Majid and Senu (2018), the proposed block method is said to be zero-stable when it fulfilled Definition 4 herewith:

*Definition 4* A multistep method is zero-stable if the modulus of the first characteristic polynomial's root in Equation (12) is not greater than one:

$$\rho(\xi) = det\left[\sum_{j=0}^{k} A_j\, \xi^{k-j}\right] = 0. \quad (12)$$

Accordingly,

$$\rho(\xi) = \left|\sum_{j=0}^{1} A_j\, \xi^{k-j}\right| = \left|A_0\xi^1 - A_1\xi^0\right| = \xi(\xi+1),$$

where the roots for the first characteristic polynomial in Equation are given by:

$$\xi(\xi+1) = 0$$
$$\xi = 0, -1.$$

If taking the modulus for the roots, $\xi$ the root will be 0 and 1. Based on Lambert (1973), the linear multistep method is claimed to be converged if both conditions for consistency and zero-stability are satisfied. Therefore, the proposed method, 1OBM3, converges.

## CONVERGENCE OF HYBRID MULTISTEP BLOCK METHOD

Convergence analysis is essential in numerical analysis since it enables us to comprehend how numerical methods behave as the number of iterations or step sizes approaches infinity. Relying on Ismail, Majid and Senu (2022), the convergence analysis for 1OBM3 is demonstrated using the following theorem, where the requirement is given as follows:

*Theorem 1* The approximate solutions of Equation (9) converge to its precise solution.

*Proof* A linear multistep method with an off-step point, Equation (9), is convergent if the following two conditions are fulfilled:

$$\lim_{h\to 0} y_{n+1} = y^*_{n+1}$$
$$\lim_{h\to 0} y_{n+2} = y^*_{n+2}\,. \quad (13)$$

From the conditions, if the suggested strategy eventually approaches the exact phrase, $y_{n+1}^{*}$ where i = 0,1,2,..., then the strategy can be utilized since it has properly converged. Using the approximate corrector of 1OBM3 shown in Equation (9), the exact solution is given by:

$$y^{*c}_{n+1} = y_n + \frac{h}{36}\left(3f_{n+1} + 32f_{n+\frac{1}{2}} + f_{n-1}\right) + \frac{32}{36}h^4 y^{(4)}(\xi_n),$$
$$y^{*c}_{n+2} = y_n + \frac{h}{9}\left(5f_{n+2} + 16f_{n+\frac{1}{2}} - 3f_n\right) + \frac{16}{9}h^4 y^{(4)}(\xi_n). \quad (14)$$

By applying the Lipschitz condition, we obtain,

$$|f(x, y^*(x)) + \int_{\alpha}^{x} K(x, y^*(x), y^*(\alpha), y'^*(\alpha)) -$$
$$f(x, y(x)) + \int_{\alpha}^{x} K(x, y(x), y(\alpha), y'(\alpha))| \leq L|y^* - y|. \quad (15)$$

The exact and approximate solutions are subtracted, eventually giving Equation (16), after letting

$$y^*_{n+1} - y_{n+1} = d_{n+1},\, y^*_{n+2} - y_{n+2} = d_{n+2},\, y^*_n - y_n = d_n,$$

$$|d_{n+1}| \le |d_n| + \frac{3}{36}hL|d_{n+1}| + \frac{32}{36}hL\left|d_{n+\frac{1}{2}}\right|$$

$$+\frac{1}{36}hL|d_{n-1}| + \frac{32}{36}h^4y^{(4)}(\xi_n), \qquad (16)$$

$$|d_{n+2}| \le |d_n| + \frac{5}{9}hL|d_{n+2}| + \frac{16}{9}hL\left|d_{n+\frac{1}{2}}\right|$$

$$-\frac{3}{9}hL|d_n| + \frac{16}{9}h^4y^{(4)}(\xi_n).$$

As $h$ approaches zero, we now have,

$$|d_{n+1}| \le |d_n| \;\Rightarrow\; y^*_{n+1} - y_{n+1} \le y^*_n - y_n$$

$$\Rightarrow\; y^*_{n+1} - y^*_n \le y_{n+1} - y_n$$

$$|d_{n+2}| \le |d_n| \;\Rightarrow\; y^*_{n+2} - y_{n+2} \le y^*_n - y_n$$

$$\Rightarrow\; y^*_{n+2} - y^*_n \le y_{n+2} - y_n.$$

As a result, it has been determined that the suggested strategy is converged since it is getting close to the exact solution. The convergence analysis is crucial for several reasons, including verifying numerical techniques. By contrasting the results of numerical approaches with the exact solutions to a problem, convergence analysis offers a mechanism to validate these solutions. Note that a numerical approach can be regarded as accurate and dependable if it converges to the true solution.

## STABILITY ANALYSIS OF HYBRID MULTISTEP BLOCK METHOD

The stability region is crucial to plot in numerical analysis because it clarifies how numerical methods behave when used to solve differential equations or other mathematical issues. A numerical method is stable for a set of parameters when it falls within the stability area, which is a region in the complex plane where the solution of the method produces does not diverge or fluctuate uncontrollably. The numerical stability is investigated in this section where a linear test equation for NDVIDE is presented below:

$$y'(x) = \xi y(qx) + \nu\int_0^{qx} y(u)du + \eta y'(qx), \qquad (17)$$

where the test equation is obtained from Wu and Gan (2008). For simplicity, assume assume that $qx = mh$ ($m \in I$) and $y(qx) = Y_{nr}$, (Rihan et al. 2009). For 1OBM3 in Equation (9) the multistep formula is rearranged as follows:

$$\sum_{j=0}^{2} A_k\, Y_{N+k} = h\sum_{j=0}^{2} B_k\, F_{N+k}. \qquad (18)$$

where the values for $A_k$ and $B_k$ are given below:

$$A_0 = \begin{bmatrix} 0 & -1 \\ 0 & -1 \end{bmatrix}, \quad A_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix},$$

$$B_0 = \begin{bmatrix} \frac{1}{36} & 0 \\ 0 & -\frac{3}{9} \end{bmatrix}, \quad B_1 = \begin{bmatrix} \frac{32}{36} & \frac{3}{36} \\ \frac{16}{9} & 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 & 0 \\ \frac{5}{9} & 0 \end{bmatrix}.$$

From Equation (17), we then obtain:

$$F_N = y'(x) = \xi y(qx) + \nu\int_0^{qx} y(u)du + \eta y'(qx),$$

$$F_N = \xi Y_{nr} + \nu\int_0^{qx} y(u)du + \eta Y'_{nr} \qquad .$$

The numerical integration is adapted into an integral part. Hence, by applying Simpson's quadrature rule, we have:

$$\int_0^x y(u)du = h\left(\frac{1}{3}Y_{N-2} + \frac{4}{3}Y_{N-1} + \frac{1}{3}Y_N\right), \qquad (19)$$

and implementing the test Equations (17) and (19) into Equation (18) yields,

$$A_0Y_N + A_1Y_{N+1} + A_2Y_{N+2}$$

$$= hB_0\left(\xi Y_{nr} + \nu h\left(\frac{1}{3}Y_{N-2} + \frac{4}{3}Y_{N-1} + \frac{1}{3}Y_N\right) + \eta Y'_{nr}\right)$$

$$+hB_0\left(\xi Y_{nr} + \nu h\left(\frac{1}{3}Y_{N-1} + \frac{4}{3}Y_N + \frac{1}{3}Y_{N+1}\right) + \eta Y'_{nr}\right)$$

$$+hB_0\left(\xi Y_{nr} + \nu h\left(\frac{1}{3}Y_N + \frac{4}{3}Y_{N+1} + \frac{1}{3}Y_{N+2}\right) + \eta Y'_{nr}\right).$$

By rearranging and substituting $H_1 = \eta h$ *and* $H_2 = \nu h^2$, the stability polynomial may be determined as follows:

$$\begin{aligned}\pi(H_1, H_2; t) = \det(&t^{r+2}\left(A_2 - \tfrac{1}{3}H_2B_2\right) \\ &+ t^{r+1}\left(A_1 - \tfrac{1}{3}H_2B_1 - \tfrac{4}{3}H_2B_2\right) \\ &+ t^{r}\left(A_0 - \tfrac{1}{3}H_2B_0 - \tfrac{4}{3}H_2B_1 - \tfrac{1}{3}H_2B_2\right) \quad (20) \\ &+ t^{r-1}\left(-\tfrac{4}{3}H_2B_0 - \tfrac{1}{3}H_2B_1\right) \\ &+ t^{r-2}\left(-\tfrac{1}{3}H_2B_0\right) + t^{3m}(-H_1B_2 - \eta B_2) \\ &+ t^{2m}(-H_1B_1 - \eta B_1) \\ &+ t^{0}(-H_1B_0 - \eta B_0)) = 0,\end{aligned}$$

The complete calculation for the determinant is computed using Maple software. Hence, the stability polynomial is given by:

$$t^5\left(-\frac{41}{108}-\frac{23}{54}H_1-\frac{23}{162}H_2-\frac{5}{162}H_1H_2-\frac{5}{108}H_1^2-\frac{5}{972}H_2^2\right)+$$
$$t^4\left(\frac{29}{27}+\frac{25}{27}H_1-\frac{29}{108}H_2-\frac{2}{9}H_1H_2-\frac{4}{27}H_1^2-\frac{14}{243}H_2^2\right)+$$
$$t^3\left(-\frac{11}{12}-\frac{11}{12}H_1+\frac{107}{162}H_2-\frac{85}{162}H_1H_2-\frac{125}{486}H_2^2\right)+$$
$$t^2\left(-\frac{8}{27}-\frac{16}{27}H_1-\frac{37}{27}H_2-\frac{16}{27}H_1H_2-\frac{8}{27}H_1^2-\frac{65}{108}H_2^2\right)+$$
$$t^1\left(\frac{1}{36}+\frac{1}{36}H_1-\frac{41}{54}H_2-\frac{1}{2}H_1H_2-\frac{239}{324}H_2^2\right)+$$
$$t^0\left(-\frac{1}{108}-\frac{1}{54}H_1-\frac{37}{324}H_2-\frac{10}{81}H_1H_2-\frac{1}{108}H_1^2-\frac{145}{486}H_2^2\right)+$$
$$t^{-1}\left(-\frac{1}{162}H_2-\frac{1}{162}H_1H_2-\frac{10}{243}H_2^2\right)+t^{-2}\left(-\frac{1}{972}H_2^2\right),$$

As a result, after setting $\eta = 1$, the stability region for the 1OBM3 is depicted in Figure 2. The stability region may recommend selecting appropriate step sizes or other numerical technique parameters to guarantee stability and accuracy.

In Figure 2, it can be observed that the regions are progressively smaller as m increases, and the step size, $h$ decreased, $(\frac{\tau}{h} = m)$ where $\tau = 1$. The regions of stability are obtained by replacing $t$ with -1,0,1 and $t = \cos\theta + i\sin\theta$ for $0 \le \theta \le 2\pi$ in stability polynomial. The stability regions are absolutely stable since all roots in the stability polynomial satisfies $|t| \le 1$.

## IMPLEMENTATION OF HYBRID MULTISTEP BLOCK METHOD

In order to solve the pantograph equation and mixed type of NVDIDE by applying 1OBM3, two approximations, including one off-point, are calculated in one block using the constant step size technique. Prior to computing the

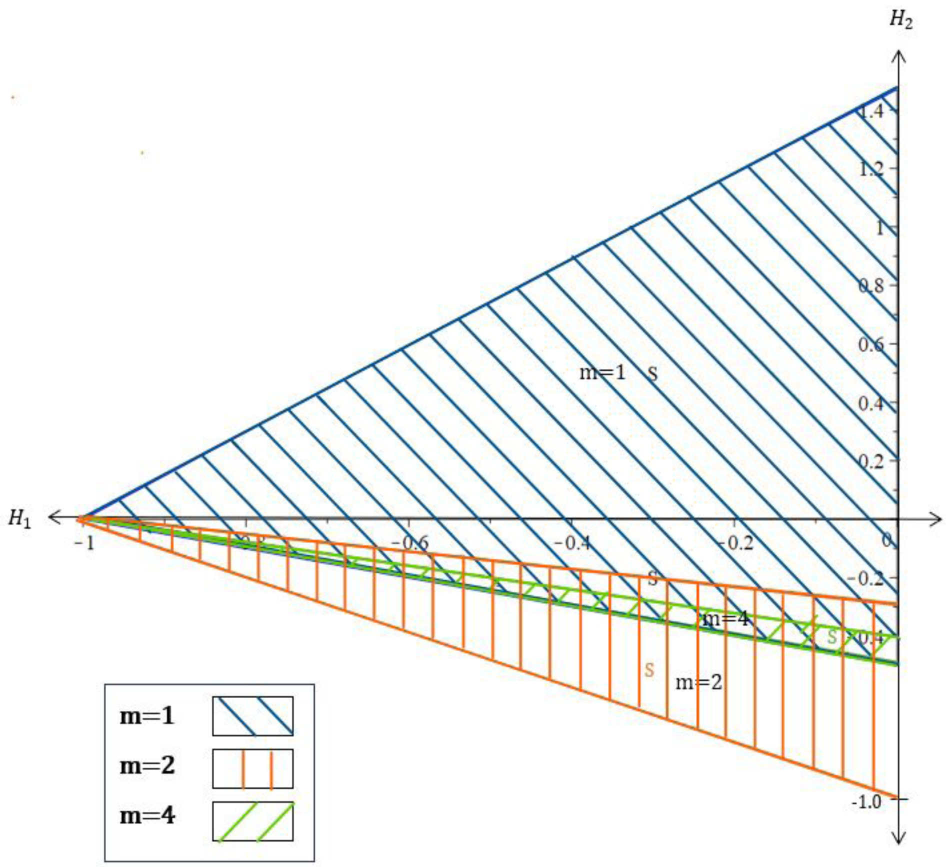


FIGURE 2. Areas of numerical stability for 1OBM3 with varying values of $m$ = 1;2;4

suggested approach, it is also necessary to determine the position of delays. The delay term and its derivative, as well as the integral and both functions of delay, are considered. In this research, third-order backward (BDF) and forward differentiation formulae (FDF) are derived and applied in solving $y'(\alpha)$. The formulae are shown as follow:

$$\begin{aligned} y'_n &= \frac{2y_n - y_{n-1} - y_{n-2}}{3h}\,(BDF) \\ y'_n &= \frac{-2y_n + y_{n+1} + y_{n+2}}{3h}\,(FDF). \end{aligned} \tag{21}$$

The Composite Simpson quadrature rule will be implemented for the integration part. Note that a realistic computation of the integral over the full interval is obtained using Simpson's rule for each subinterval, as presented below:

$$\begin{aligned} \int_a^b f(x)dx &\approx \frac{h}{3}\sum_{j=1}^{\frac{n}{2}}\left[f(x_{2j-2}) + 4f(x_{2j-1}) + f(x_{2j})\right] \\ &= \frac{h}{3}\left[f(x_0) + 2\sum_{j=1}^{\frac{n}{2}-1} f(x_{2j}) + 4\sum_{j=1}^{\frac{n}{2}} f(x_{2j-1}) + f(x_n)\right] \end{aligned} \tag{22}$$

where $x_j = a + jh$ for $j = 0,1,\ldots, n-1, n,$ with $h = \frac{b-a}{n}$ in particular, $x_0 = a$ and $x_n = b$. In solving the delay problem, if the delay is smaller than the initial point, the initial function is evaluated. If the delay is larger than the initial point, an additional derived method is applied to calculate the pantograph equation,

$$y_{n+\frac{1}{2}} = y_n + \frac{h}{2}(f_n) \tag{23}$$

while a Lagrange interpolating polynomial is applied to solve the mixed delay problem.

The initial solutions must be taken into consideration prior to implementing 1OBM3. Consequently, a single-step method is employed since the hybrid multistep block method cannot be implemented alone. Three initial solutions are approximated for 1OBM3 since the predictor formula is of order three. The one-step technique for 1OBM3 merely computes two approximated solutions since the starting value is already provided. To evaluate the initial solutions for the proposed method, a Runge-Kutta of order three (RK3) was considered. The formula of RK3 is shown below:

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{6}(k_1 + 4k_2 + k_3) \\ k_1 &= f\left(x_n, y_n + \int_\alpha^n K(x_n, y_n, y_\alpha, y'_\alpha)\right) \\ k_2 &= f\left(x_n + \frac{h}{2}, \left(y_n + \frac{h}{2}k_1\right) + \int_\alpha^n K(x_n, y_n, y_\alpha, y'_\alpha)\right) \\ k_3 &= f\left(x_n + h, (y_n + 2hk_2 - hk_1) + \int_\alpha^n K(x_n, y_n, y_\alpha, y'_\alpha)\right). \end{aligned} \tag{24}$$

All proportional and mixed delay problem-solving algorithms have been developed using the constant step size technique in the C programming language. Here, the numerical outcomes demonstrated the applicability and effectiveness of the suggested methods.

## ALGORITHM OF HYBRID MULTISTEP BLOCK METHOD

This subsection illustrates of the 1OBM3 algorithm for the pantograph equation and mixed delay problem. The algorithms are displayed in detail on how to handle the integral part, the delay term, and its derivative. The following notations are used in the algorithm: $a$ = Initial value, $b$ = End value, $h$ Step size, $N$ Number of iterations, $y_0$ = Initial solution, $y'(\alpha)$ = Delay derivative for mixed delay, and $y'(qx)$ = Delay derivative for proportional delay.

## ALGORITHM OF 1OBM3 FOR PANTOGRAPH EQUATION

**Step 1**: All values given in equations, $x_0$ = a, $x_n = b$, $h$, $N$, $y_0$, $y'(qx) \leq a$ are set. **Step 2**: The pantograph equation for NDVIDE is given in Equation (2). **Step 3**: The delay terms are solved by applying a prior solution if $qx \geq a$. **Step 4**: The delay terms are resolved by utilizing an additional derived method in Equation (23) if $qx \geq$ a. **Step 5**: Backward or forward divided difference formulae are tested to find $y'(qx)$.**Step 6**: Composite Simpson is applied to approximate the integral part. **Step 7**: For $n$ = 0,1. The initial solution is computed by applying RK3 denoted in Equation (24). **Step 8**: For $n$ = 2, 4, 6,… Approximate NDVIDE by using the proposed method, 1OBM3, denoted in Equation (9). **Step 9**: Average and maximum error, total steps and function calls are calculated computationally. **Step 10**: Stop.

## ALGORITHM OF 1OBM3 FOR MIXED DELAY

**Step 1**: All values given in equations $x_0$ = a, $x_n = b$, $h$, $N$, $y_0$, $y'(qx) \leq a$ are set. **Step 2**: The mixed delay for NDVIDE is given in Equation (1). **Step 3**: The original function given is used if $\alpha \leq a$. **Step 4**: The delay terms are resolved using Lagrange interpolating polynomial if $\alpha > a$. **Step 5**: Backward or forward divided difference formulae are tested to find $y'(\alpha)$. **Step 6**: Composite Simpson is applied to approximate the integral part. **Step 7**: For $n$ = 0,1. The initial solution is computed by applying RK3 denoted in Equation (24). **Step 8**: For $n$ = 2,4,6,… Approximate NDVIDE using the proposed method, 1OBM3, denoted in Equation (9). **Step 9**: Average and maximum error, total steps and function calls are calculated computationally. **Step 10**: Stop.

## NUMERICAL RESULTS

In this section, first-order NDVIDE problems with mixed and proportional delays are considered. Two tested problems of proportional delay and two tested problems of mixed delays are solved numerically using the two-point one off-step point hybrid multistep block method, 1OBM3. Other specific classes of Volterra integro-differential equations, such as those where the kernel is independent of $y'(t)$ can also be solved using the suggested method with efficiency. Note that all tested examples were solved with different values of constant step size. The numerical results for Examples 1 - 4 are shown in Tables 1 - 4. By creating two approximations in a single step, one of which comprises the off-point, all the tested problems were numerically resolved using the C programming language. Consequently, the accuracy, total function evaluations, and the number of steps are compared between the proposed method, 1OBM3 and 2MVIDE3 from Mohamed and Majid (2016), Adam Moulton and RK3. The notations used in the tables are as follows: $h$ = Step size, MTD = Method, FC = Total function calls, TSTEP = Total Step, MERR = Maximum Error, AERR = Average Error, 1OBM3 = Two-point one off-step point hybrid multistep block method (Order 3), 2MVIDE3 = Two-point Fully Implicit Block Method (Order 3), ABM3 = Adam-Bashforth-Moulton Method (Order 3), RK3 = Runge Kutta Method (Order 3), $5e-10$ = $5 \times 10^{-10}$.

Example 1

$$y'(x) = -sin\left(\frac{x}{4}\right) cos\left(\frac{x}{4}\right) + \int_0^{\frac{x}{4}} [-sin(t)y(t) + cos(t)y'(t)]dt + \cos(\mathrm{x})$$

$$y(0) = 0$$

Exact solution:

$$y(x) = sin(x), \qquad x \in [0,1]$$

*Example 2*

$$y'(x) = y'\left(\frac{x}{2}\right) + xcos(x) - \sin(x) - \frac{\cos\left(\frac{x}{2}\right)}{2} + \int_0^{\mathrm{x}} [ty(t)]dt + cos(x)$$

$$y(0) = 0$$

Exact solution:

$$y(x) = sin(x), \qquad x \in [0,1]$$

*Example 3*

$$y'(x) = y'(x-1) - \frac{x^2}{8} - \frac{x}{2} + \int_0^{\frac{x}{2}} y(t)\, dt,$$

$$y(0) = 1,$$

Exact solution:

$$y(x) = x + 1, \qquad x \in [0,1]$$

*Example 4*

$$y'(x) = y\,(x) + y(x-1) - e^{\frac{x}{2}}x^2 + e^{\frac{x}{2}}x - e^{x-1} + \int_0^{x} [x\, e^{-t} y(t)]dt + \int_0^{\frac{x}{2}} [(x^2 - 2t - 2)\, y'(t)]dt,$$

$$y(0) = 1,$$

Exact solution:

$$y(x) = e^x, \qquad x \in [0,1]$$

TABLE 1. Numerical result for Example 1

| *h* | MTD | FC | TSTEP | MERR | AERR |
|---|---|---|---|---|---|
| 0.1 | 1OBM3 | 6 | 6 | 1.4130e-02 | 5.6300e-03 |
| | 2MVIDE3 | 10 | 6 | 3.7027e-02 | 1.4083e-02 |
| | ABM3 | 10 | 10 | 9.7075e-03 | 4.5462e-03 |
| | RK3 | 30 | 10 | 1.6173e-02 | 8.8926e-03 |
| 0.01 | 1OBM3 | 51 | 51 | 1.5187e-03 | 4.8531e-04 |
| | 2MVIDE3 | 100 | 51 | 2.3798e-03 | 8.1800e-04 |
| | ABM3 | 100 | 100 | 6.9888e-04 | 3.0623e-04 |
| | RK3 | 300 | 100 | 1.6198e-03 | 7.5410e-04 |
| 0.001 | 1OBM3 | 501 | 501 | 1.7547e-04 | 6.2421e-05 |
| | 2MVIDE3 | 1000 | 501 | 2.2381e-04 | 7.8616e-05 |
| | ABM3 | 1000 | 1000 | 7.8904e-05 | 3.9663e-05 |
| | RK3 | 3000 | 1000 | 1.6172e-04 | 7.3772e-05 |
| 0.0001 | 1OBM3 | 5001 | 5001 | 1.7780e-05 | 6.3987e-06 |
| | 2MVIDE3 | 10000 | 5001 | 2.2238e-05 | 7.8416e-06 |
| | ABM3 | 10000 | 10000 | 7.9804e-06 | 4.0690e-06 |
| | RK3 | 30000 | 10000 | 1.6170e-05 | 7.3606e-06 |

TABLE 2. Numerical result for Example 2

| $h$ | MTD | FC | TSTEP | MERR | AERR |
|---|---|---|---|---|---|
| 0.1 | 1OBM3 | 6 | 6 | 2.0388e-02 | 1.6841e-02 |
| | 2MVIDE3 | 10 | 6 | 3.2477e-02 | 1.1130e-02 |
| | ABM3 | 10 | 10 | 2.1625e-02 | 1.8382e-02 |
| | RK3 | 30 | 10 | 2.0388e-02 | 1.6841e-02 |
| 0.01 | 1OBM3 | 51 | 51 | 2.7278e-03 | 1.0101e-03 |
| | 2MVIDE3 | 100 | 51 | 3.3276e-03 | 1.1848e-03 |
| | ABM3 | 100 | 100 | 2.0747e-03 | 1.5197e-03 |
| | RK3 | 300 | 100 | 2.1180e-03 | 1.5850e-03 |
| 0.001 | 1OBM3 | 501 | 501 | 2.7729e-04 | 1.0220e-04 |
| | 2MVIDE3 | 1000 | 501 | 3.3328e-04 | 1.1952e-04 |
| | ABM3 | 1000 | 1000 | 2.0823e-04 | 1.4907e-04 |
| | RK3 | 3000 | 1000 | 2.1284e-04 | 1.5742e-04 |
| 0.0001 | 1OBM3 | 5001 | 5001 | 2.7773e-05 | 1.0232e-05 |
| | 2MVIDE3 | 10000 | 5001 | 3.3333e-05 | 1.1963e-05 |
| | ABM3 | 10000 | 10000 | 2.0832e-05 | 1.4879e-05 |
| | RK3 | 30000 | 10000 | 2.1295e-05 | 1.5731e-05 |

TABLE 3. Numerical result for Example 3

| $h$ | MTD | FC | TSTEP | MERR | AERR |
|---|---|---|---|---|---|
| 0.1 | 1OBM3 | 6 | 6 | 4.9074e-02 | 1.3583e-02 |
| | 2MVIDE3 | 10 | 6 | 3.1375e-02 | 1.1338e-02 |
| | ABM3 | 10 | 10 | 8.9038e-02 | 6.4965e-02 |
| | RK3 | 30 | 10 | 4.4840e-01 | 1.5802e-01 |
| 0.01 | 1OBM3 | 51 | 51 | 7.3068e-03 | 1.8207e-03 |
| | 2MVIDE3 | 100 | 51 | 4.1762e-03 | 1.4433e-03 |
| | ABM3 | 100 | 100 | 8.4203e-03 | 6.5668e-03 |
| | RK3 | 300 | 100 | 5.9443e-01 | 2.0952e-02 |
| 0.001 | 1OBM3 | 501 | 501 | 7.4806e-04 | 1.8693e-04 |
| | 2MVIDE3 | 1000 | 501 | 4.4174e-04 | 1.4766e-04 |
| | ABM3 | 1000 | 1000 | 8.3423e-04 | 6.5630e-04 |
| | RK3 | 3000 | 1000 | 6.0944e-01 | 2.1470e-03 |
| 0.0001 | 1OBM3 | 5001 | 5001 | 7.4978e-05 | 1.8743e-05 |
| | 2MVIDE3 | 10000 | 5001 | 4.4417e-05 | 1.4800e-05 |
| | ABM3 | 10000 | 10000 | 8.3342e-05 | 6.5625e-05 |
| | RK3 | 30000 | 10000 | 6.1094e-01 | 2.1522e-04 |

TABLE 4. Numerical result for Example 4

| $h$ | MTD | FC | TSTEP | MERR | AERR |
|---|---|---|---|---|---|
| 0.1 | 1OBM3 | 6 | 6 | 1.0645e-01 | 3.6297e-02 |
| | 2MVIDE3 | 10 | 6 | 1.4810e-01 | 8.3027e-02 |
| | ABM3 | 10 | 10 | 3.1559e-01 | 1.9323e-01 |
| | RK3 | 30 | 10 | 7.8390e-01 | 4.7050e-01 |
| 0.01 | 1OBM3 | 51 | 51 | 1.5209e-02 | 5.7137e-03 |
| | 2MVIDE3 | 100 | 51 | 1.7762e-02 | 1.1128e-02 |
| | ABM3 | 100 | 100 | 3.5733e-02 | 2.2538e-02 |
| | RK3 | 300 | 100 | 3.9392e-01 | 5.6257e-02 |
| 0.001 | 1OBM3 | 501 | 501 | 1.5770e-03 | 5.9636e-04 |
| | 2MVIDE3 | 1000 | 501 | 1.8086e-03 | 1.1422e-03 |
| | ABM3 | 1000 | 1000 | 3.6192e-03 | 2.2873e-03 |
| | RK3 | 3000 | 1000 | 3.6752e-01 | 5.7224e-03 |
| 0.0001 | 1OBM3 | 5001 | 5001 | 1.5826e-04 | 5.9890e-05 |
| | 2MVIDE3 | 10000 | 5001 | 1.8118e-04 | 1.1452e-04 |
| | ABM3 | 10000 | 10000 | 3.6239e-04 | 2.2907e-04 |
| | RK3 | 30000 | 10000 | 3.6498e-01 | 5.7321e-04 |

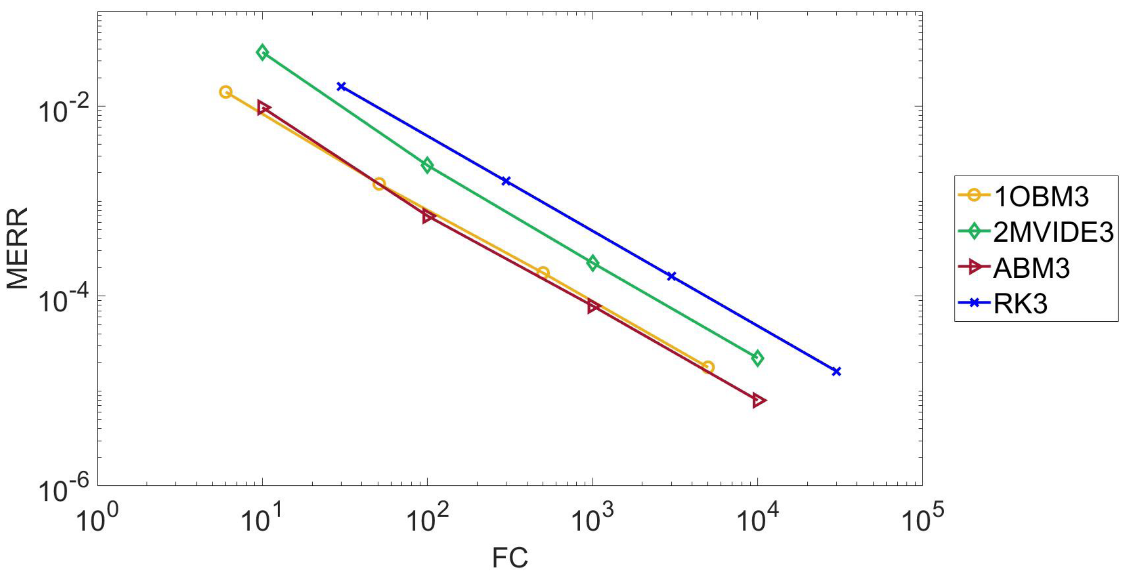


FIGURE 3. MERR vs. FC for Table 1

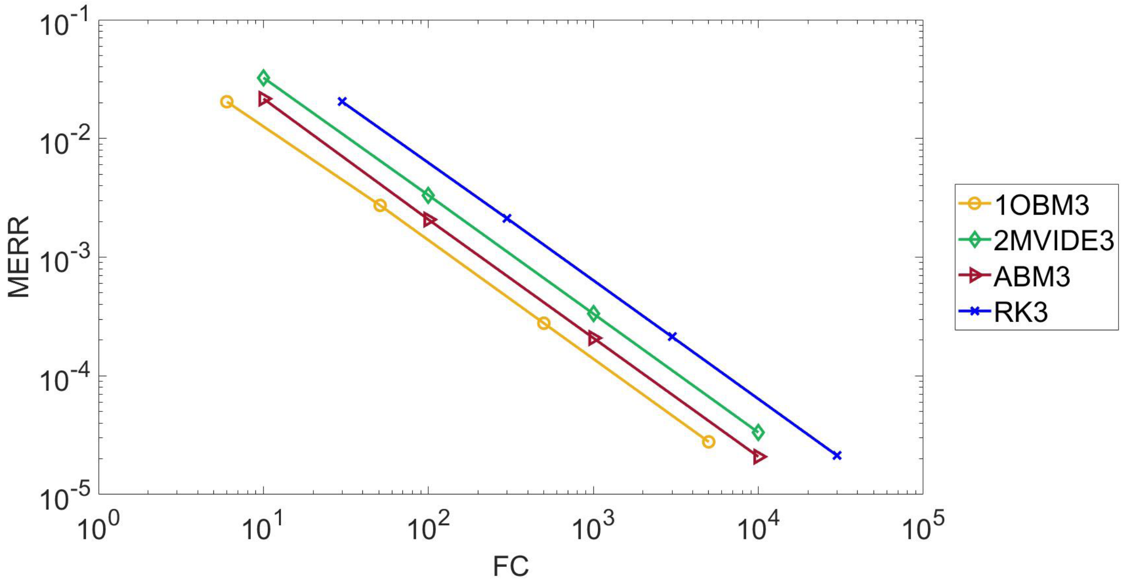


FIGURE 4. MERR vs. FC for Table 2

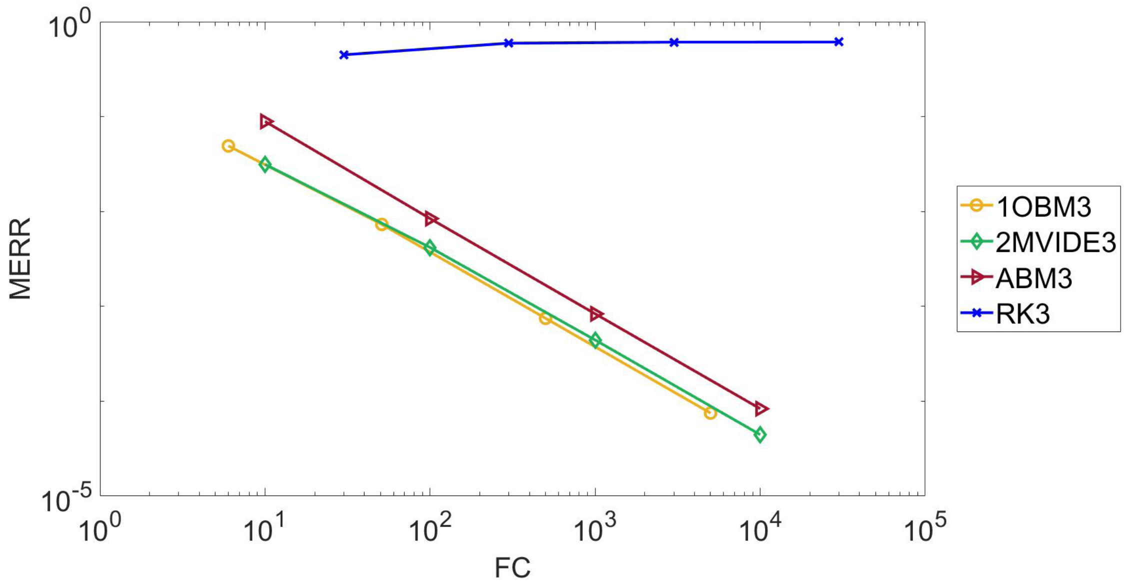


FIGURE 5. MERR vs. FC for Table 3

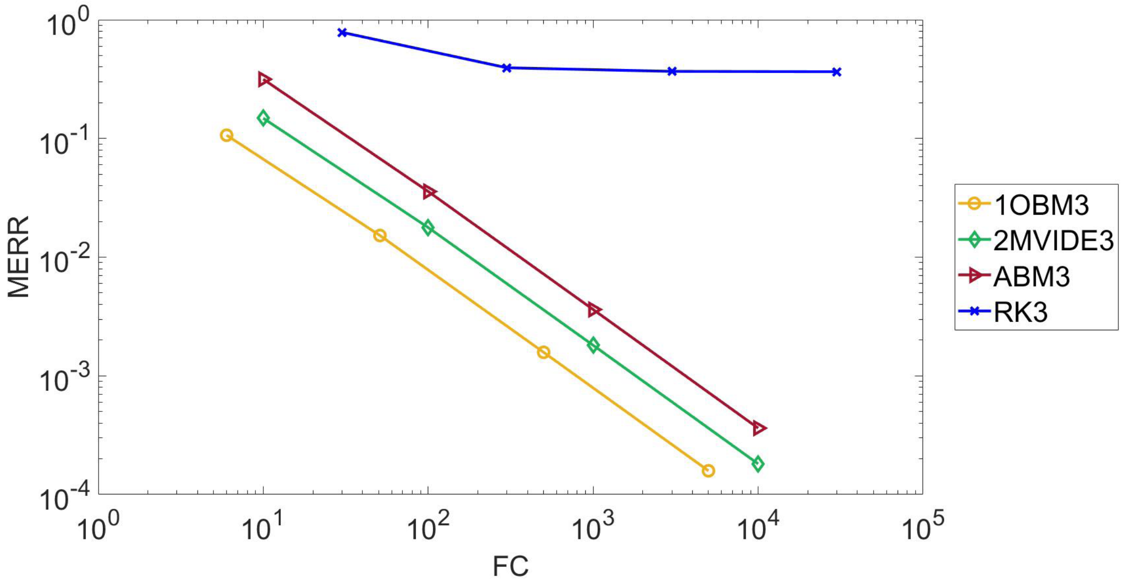


FIGURE 6. MERR vs. FC for Table 4

## DISCUSSION

From the numerical result obtained in Table 1, the AERR and MERR for 1OBM3 at h = 0.1,0.01,0.001 and 0.0001 is comparable with 2MVIDE3 and RK3. The results for ABM3 is slightly better but the FC and TSTEP needed is higher than 1OPBM3. Hence, they are also comparable. Additionally, the FC for 1OBM3 is lesser than 2MVIDE3 and RK3 since the proposed method computed all solutions in a single step. Even though both 1OBM3 and 2MVIDE3 are block methods, the FC for 1OBM3 are lesser than 2MVIDE3. Meanwhile, the proposed 1OBM3 is superior to RK3 in terms of TSTEP. Since highlighting the benefits of an implicit hybrid block method is also the primary priority of the discussion, all the factors that affect how block and non-block methods are compared must be considered.

In Table 2, the AERR and MERR for 1OBM3 were found to be compatible with 2MVIDE3, ABM3 and RK3 at all step sizes. 1OBM3 outperformed 2MVIDE3, ABM3 and RK3 in terms of FC. Since 1OBM3 is evaluated in a block, fewer steps are necessary overall than for the other methods (ABM3 and RK3). In Table 3, the approximate solutions for 1OBM3 are comparable with 2MVIDE3 and ABM3 as the step size is smaller. The FC for 1OBM3 is lesser than 2MVIDE3 taken from Mohamed and Majid (2016), even though 2MVIDE3 is also a block method.

Finally, in Table 4, the proposed method is comparable to 2MVIDE3 and ABM3, except for the AERR at h=0.01, 0.001, 0.0001 where the maximum error for 1OBM3 is more accurate than RK3. Although the proposed solution appears equivalent, it calls fewer functions and requires fewer steps than the other methods. In conclusion, compared to 2MVIDE3, ABM3 and RK3, the proposed 1OBM3 provides accurate results on several parameters such as the accuracy, the total number of steps required and the evaluated function call. Additionally, an implicit hybrid method outperformed the other ones in accuracy by computing the numerical solutions via simultaneously estimating several points, including the off-point in the predictors. Therefore, the proposed methods are suitable for solving NDVIDE with mixed delays and pantograph equations.

## CONCLUSION

NDVIDE was numerically treated with mixed delays and pantograph equation in this study. Consequently, NDVIDE problems were resolved using 1OBM3, which produced two approximations in a single step using the identical prior data, including the off-point. Compared to ABM3, RK3, and 2MVIDE3, the number of overall steps and the function evaluated when solving NDVIDE problems can be decreased using the suggested approach. The numerical data showed that the maximum and average errors decreased with decreasing step size. An assortment of examples is given to highlight the value of the recommended strategy, and it is noted that they all perfectly match the exact solutions. Additionally, the numerical results show that 1OBM3 effectively handles NDVIDE problems involving mixed delays and the pantograph equation.

## ACKNOWLEDGEMENTS

All authors gratefully acknowledge the financial support of the Fundamental Research Grant Scheme (project code: FRGS/1/2020/STG06/UPM/01/1) from the Ministry of Higher Education Malaysia. The authors are also thankful to the referees for their useful comments and suggestions.

## REFERENCES

Altun, Y. 2021a. Asymptotic behaviours of the solutions of neutral type Volterra integro-differential equations and some numerical solutions via differential transform method. *MANAS Journal of Engineering* 9(Special 1): 49-57.

Altun, Y. 2021b. A new result on the global exponential stability of nonlinear neutral Volterra integro-differential equation with variable lags. *Mathematics in Natural Science* 5(2019): 29-43.

Baharum, N.A., Majid, Z.A. & Senu, N. 2018. Solving Volterra integrodifferential equations via diagonally implicit multistep block method. *International Journal of Mathematics and Mathematical Sciences* 2018: 7392452.

Gurbuz, B. 2021. A numerical scheme for the solution of neutral integro-differential equations including variable delay. *Mathematical Sciences* 16(1-9): 13-21.

Ismail, N.I.N., Majid, Z.A. & Senu, N. 2022. Numerical solution for neutral delay differential equation of constant or proportional type using hybrid block method. *Advanced in Applied Mathematics and Mechanics* 14(5): 1138-1160.

Ismail, N.I.N., Majid, Z.A. & Senu, N. 2020. Solving neutral delay differential equation of pantograph type. *Malaysian Journal of Mathematical Sciences (ICoAIMS2019)* 14(S): 107-121.

Jator, S.N. 2010. On the hybrid method with three off-step points for initial value problems. *International Journal of Mathematical Education in Science and Technology* 41(1): 110-118.

Laib, H., Bellour, A. & Boulmerka, A. 2022. Taylor collocation method for high-order neutral delay Volterra integro-differential equations. *Journal of Innovative Applied Mathematics and Computational Sciences* 2(1): 43-77.

Lambert, J.D. 1991. *Numerical Methods for Ordinary Differential Systems.* New York: John Wiley & Sons.

Lambert, J.D. 1973. *Computational Methods in Ordinary Differential Equations*. New York: John Wiley & Sons.

Li, Y. & Li, S. 2021. Classical theory of linear multistep methods for Volterra functional differential equations. *Discrete Dynamics in Nature and Society* 2021: 1-15.

Mirzaee, F., Bimesl, S. & Tohidi, E. 2016. A numerical framework for solving high-order pantograph-delay Volterra integro-differential equations. *Kuwait Journal of Science* 43(1): 69-83.

Mohamed, N.A. & Majid, Z.A. 2016. Multistep block method for solving Volterra integro-differential equations. *Malaysian Journal of Mathematical Sciences* 10(2016): 33-48.

Reutskiy, S.Y. 2016. The backward substitution method for multipoint problems with linear volterra–fredholm integro-differential equations of the neutral type. *Journal of Computational and Applied Mathematics* 296: 724-738.

Rihan, F.A., Doha, E.H., Hassan, M.I. & Kamel, N. 2009. Numerical treatments for Volterra delay integro-differential equations. *Computational Methods in Applied Mathematics* 9(3): 292-318.

Vijayakumar, V. 2018, Approximate controllability results for abstract neutral integrodifferential inclusions with infinite delay in Hilbert spaces. *IMA Journal of Mathematical Control and Information* 5(1): 297-314.

Wen, L. & Zhou, Y. 2017. Convergence of one-leg methods for neutral delay integro-differential equations. *Journal of Computational and Applied Mathematics* 317: 432-446.

Wen, L. & Yu, Y. 2016. Convergence of Runge–Kutta methods for neutral delay integro-differential equations. *Applied Mathematics and Computation* 282(2016): 84-96.

Wu, S. & Gan, S. 2008. Analytical and numerical stability of neutral delay integro-differential equations and neutral delay partial differential equations. *Computers & Mathematics with Applications* 55(11): 2426-2443.

Yuzbasi, S. & Karacayir, M. 2017. A numerical approach for solving high-order linear delay Volterra integro-differential equations. *International Journal of Computational Methods* 15(5): 1850042.

*Corresponding author; email: am_zana@upm.edu.my