Malaysian Journal of Mathematical Sciences 15(3): 369–385 (2021)

Malaysian Journal of Mathematical Sciences

Journal homepage: https://einspem.upm.edu.my/journal

Numerical Computation of Third Order Delay Differential Equations by Using Direct Multistep Method

Jaaffar, N. T.[1], Majid, Z. A.*[1,2], and Senu, N.[1,2]

[1]*Institute for Mathematical Research, Universiti Putra Malaysia, Malaysia*
[2]*Department of Mathematics & Statistics, Faculty of Science, Universiti Putra Malaysia, Malaysia*

E-mail: am_zana@upm.edu.my
**Corresponding author*

Received: 29 October 2020
Accepted: 22 June 2021

Abstract

This paper introduces a direct multistep method to solve third order delay differential equations (DDEs) based on the boundary conditions given. The multistep method is presented in direct integration approach to reduce the total function calls involved and the method is derived implicitly so that the accuracy is attained. The method is also in block for every iteration to reduce total steps taken. The DDEs involve the endpoints of boundary conditions, hence, the shooting technique is to choose for the best value of additional initial value. The constant and pantograph delay types are the DDEs problems considered in this study. Lagrange interpolation is used to interpolate the delay involved in pantograph problems. The observation of the multistep method in terms of order, consistency, and convergence is also presented in this paper. The numerical results obtained are compared with the previous multistep method to verify the capability of the proposed method to solve third order DDEs directly.

Keywords: boundary value problem; delay differential equations; multistep method; shooting technique.

1 Introduction

The mathematical models play such an important role to portray the dynamic and simulation of real problems in science and engineering such as SIR model for infectious diseases, Navier-Stokes equations for fluid dynamics and Lotka-Volterra model for predator-prey interactions. Delay differential equations (DDE) is a mathematical model for the simulation of delay present in real-life problems for example delay in maturation period of species, time delay in control circuits, delay in cell division time and delay in the body's reaction to carbon dioxide. DDEs in mathematics are defined as the differential equations that associate with past and present times that are assorted in two different types of delay named as constant delay and pantograph delay. The general third order form of DDE for the constant delay is defined as below:

$$y''' = f\big(x, y(x), y'(x), y(x-\tau), y'(x-\tau)\big), \quad x \in [a, b], \tag{1}$$

$$y(x) = \phi(x), \quad y'(x) = \phi'(x), \quad x \in [a-\tau, a], \quad \tau \in \Re^{+}, \tag{2}$$

where $\phi(x)$ is the smooth initial function and τ is a positive constant for delay term where $\tau = mh$, m is a positive integer and h is the step size. Meanwhile, the general third order form of DDE for pantograph delay is defined as follows:

$$y''' = f\big(x, y(x), y'(x), y(qx), y'(qx)\big), \quad x \in [a, b], \tag{3}$$

where q is a constant that satisfies $0 < q < 1$, subject to three types of boundary conditions:

1. Type I: $y(a) = \alpha, \quad y'(a) = \gamma, \quad y(b) = \beta,$
2. Type II: $y(a) = \alpha, \quad y'(a) = \gamma, \quad y'(b) = \beta,$
3. Type III: $y(a) = \alpha, \quad y''(a) = \gamma, \quad y(b) = \beta.$

The existence and uniqueness of the solution (please refer [7]) can be studied by considering Eq. (1)-(2) with

$$\phi \in C^{r-2}[a,b], r > 2, f : [a,b] \times C^1[a,b] \times C^1[a,b] \times C[a,b] \times C[a,b] \to \Re.$$

H_1: For any $y \in C^1[a-\tau, b]$ the mapping $x \to f\big(x, y, z, u, w\big)$ is a continuous on $[a, b]$.

H_2: The following Lipchitz condition holds:

$$\begin{aligned}\big|\big|f\big(x, y_1, z_1, u_1, w_1\big) - f\big(x, y_2, z_2, u_2, w_2\big)\big|\big| \le L\big(&||y_1 - y_2||_{[a-\tau,x]} + ||z_1 - z_2||_{[a-\tau,x]} \\ &+ ||u_1 - u_2||_{[a-\tau,x-\tau]} + ||w_1 - w_2||_{[a-\tau,x-\tau]}\big),\end{aligned}$$

with $L \ge 0,\ \tau > 0$, for any $x \in [a, b]$ where $y_1, y_2, z_1, z_2 \in C^1[a, b]$ and $u_1, u_2, w_1, w_2 \in C[a, b]$.

Under the condition of H_1 and H_2, Eq.(1) has a unique solution as follows:

$$y \in C^2[a,b] \cap C^1[a-\tau, b].$$

The research began with [5] has applied Euler's method to solve numerically the second order DDEs subject to the boundary conditions. Later, other methods have been explored to solve second order DDEs such as the finite differences, cubic splines, collocation method, trapezoidal rule, Richardson extrapolation, as well as the analytical methods. Apart from that, various studies have been carried out on numerically solving the third order DDEs with initial conditions, as mentioned below.

An analytical method was developed by [1], which was the Adomian decomposition method to solve constant and pantograph delay types of third order DDEs with initial conditions. [10] employed Taylor polynomials for pantograph delay approximation. Subsequently, [6] adopted the analytical method of variational iteration and they managed to obtain better accuracy compared to the existing methods in [1] and [10].

The iterative decomposition method was implemented by [11] to solve pantograph delay type of third order DDEs. Their method of decomposition did not enforce the Adomian polynomials calculation. After that, a new algorithm was studied by [12] to convert the problem of the third order pantograph delay into the Pade approximation series. This series was a convergent series to obtain the exact solutions to the problems.

The multistep method also has been used to solve problems with DDE subject to the initial conditions such as [9] used direct Adams-Moulton method to solve second order DDEs. Then, [8] extended those methods to the direct two points block mutistep method and managed to obtain better accuracy when compared with the two and three points one-step block method. [3] implemented two points explicit multistep method to solve first order DDEs of neutral type and the accuracy obtained was comparable to the implicit method and lesser function calls needed.

Our aim in this research is to develop a direct block multistep method for solving third order DDEs with boundary conditions. The literature mentioned above were solving third order DDEs with initial conditions and there was none of block multistep method that has been studied to solve these problems with boundary conditions. Therefore, this is the contribution of the research in this paper.

2 Derivation of Method

Since the DDEs are the types of ordinary differential equations (ODEs), therefore, it is worth to consider deriving the method based on the third order ODEs:

$$y''' = f(x, y, y', y''). \tag{4}$$

The approach is to derive the method chosen to be in two points block which are y_{i+1} and y_{i+2} on the interval $[a, b]$. The third order ODEs (4) will be integrated on both sides for every two points.

First integration: y_{i+1}

$$\int_{x_i}^{x_{i+1}} y''' dx = \int_{x_i}^{x_{i+1}} f\big(x, y, y', y''\big) dx,$$

$$y''_{i+1} = y''_i + \int_{x_i}^{x_{i+1}} f\big(x, y, y', y''\big) dx. \tag{5}$$

First integration: y_{i+2}

$$\int_{x_i}^{x_{i+2}} y''' dx = \int_{x_i}^{x_{i+2}} f\big(x, y, y', y''\big) dx,$$

$$y''_{i+2} = y''_i + \int_{x_i}^{x_{i+2}} f\big(x, y, y', y''\big) dx. \tag{6}$$

Second integration: y_{i+1}

$$\int_{x_i}^{x_{i+1}} \int_{x_i}^{x} y''' dx dx = \int_{x_i}^{x_{i+1}} \int_{x_i}^{x} f\big(x, y, y', y''\big) dx dx,$$

$$y'_{i+1} = y'_i + h y''_i + \int_{x_i}^{x_{i+1}} (x_{i+1} - x) f\big(x, y, y', y''\big) dx. \tag{7}$$

Second integration: y_{i+2}

$$\int_{x_i}^{x_{i+2}} \int_{x_i}^{x} y''' dx dx = \int_{x_i}^{x_{i+2}} \int_{x_i}^{x} f\big(x, y, y', y''\big) dx dx,$$

$$y'_{i+2} = y'_i + 2h y''_i + \int_{x_i}^{x_{i+2}} (x_{i+2} - x) f\big(x, y, y', y''\big) dx. \tag{8}$$

Third integration: y_{i+1}

$$\int_{x_i}^{x_{i+1}} \int_{x_i}^{x} \int_{x_i}^{x} y''' dx dx dx = \int_{x_i}^{x_{i+1}} \int_{x_i}^{x} \int_{x_i}^{x} f\big(x, y, y', y''\big) dx dx dx,$$

$$y_{i+1} = y_i + h y'_i + \frac{h^2}{2} y''_i + \int_{x_i}^{x_{i+1}} \frac{(x_{i+1} - x)^2}{2} f\big(x, y, y', y''\big) dx. \tag{9}$$

Third integration: y_{i+2}

$$\int_{x_i}^{x_{i+2}} \int_{x_i}^{x} \int_{x_i}^{x} y''' dx dx dx = \int_{x_i}^{x_{i+2}} \int_{x_i}^{x} \int_{x_i}^{x} f\big(x, y, y', y''\big) dx dx dx,$$

$$y_{i+2} = y_i + 2h y'_i + 2h^2 y''_i + \int_{x_i}^{x_{i+2}} \frac{(x_{i+2} - x)^2}{2} f\big(x, y, y', y''\big) dx. \tag{10}$$

The function $f\big(x, y, y', y''\big)$ in Eq. (5)–(10) is approximated by using Lagrange interpolation polynomial as below:

$$P_{p,q}(x) = \sum_{j=0}^{q} \left(\prod_{n=0, n \neq j}^{q} \frac{x - x_{i+p-n}}{x_{i+p-j} - x_{i+p-n}} \right) f_{i+p-j}, \quad p = 1, 2,$$

where p is referred to as the points in the two points block i.e the first point and second point, while q is the degree of the Lagrange interpolation. Therefore, the Lagrange interpolation of degree 3 was used to interpolate the points of $\{x_{i+1}, x_i, x_{i-1}, x_{i-2}\}$ for y_{i+1} is defined as below:

$$\begin{aligned} P_{1,3}(x) = &\frac{(x - x_i)(x - x_{i-1})(x - x_{i-2})}{(x_{i+1} - x_i)(x_{i+1} - x_{i-1})(x_{i+1} - x_{i-2})} f_{i+1} + \frac{(x - x_{i+1})(x - x_{i-1})(x - x_{i-2})}{(x_i - x_{i+1})(x_i - x_{i-1})(x_i - x_{i-2})} f_i \\ &+ \frac{(x - x_{i+1})(x - x_i)(x - x_{i-2})}{(x_{i-1} - x_{i+1})(x_{i-1} - x_i)(x_{i-1} - x_{i-2})} f_{i-1} + \frac{(x - x_{i+1})(x - x_i)(x - x_{i-1})}{(x_{i-2} - x_{i+1})(x_{i-2} - x_i)(x_{i-2} - x_{i-1})} f_{i-2}. \end{aligned}$$

Meanwhile, the Lagrange interpolation of degree 3 were used to interpolate the points of $\{x_{i+2}, x_{i+1}, x_i, x_{i-1}\}$ for y_{i+2} is defined as below:

$$P_{2,3}(x) = \frac{(x-x_{i+1})(x-x_i)(x-x_{i-1})}{(x_{i+2}-x_{i+1})(x_{i+2}-x_i)(x_{i+2}-x_{i-1})}f_{i+2} + \frac{(x-x_{i+2})(x-x_i)(x-x_{i-1})}{(x_{i+1}-x_{i+2})(x_{i+1}-x_i)(x_{i+1}-x_{i-1})}f_{i+1}$$
$$+ \frac{(x-x_{i+2})(x-x_{i+1})(x-x_{i-1})}{(x_i-x_{i+2})(x_i-x_{i+1})(x_i-x_{i-1})}f_i + \frac{(x-x_{i+2})(x-x_{i+1})(x-x_i)}{(x_{i-1}-x_{i+2})(x_{i-1}-x_{i+1})(x_{i-1}-x_i)}f_{i-1}.$$

The unknown x in $P_{1,3}(x)$ and $P_{2,3}(x)$ are assumed to be $x = x_{i+2} + sh$. Then, $P_{1,3}(s)$ and $P_{2,3}(s)$ will be rewrite as follows:

$$\begin{aligned}
P_{1,3}(s) &= \frac{(2+s)(3+s)(4+s)}{6}f_{i+1} - \frac{(1+s)(3+s)(4+s)}{2}f_i + \frac{(1+s)(2+s)(4+s)}{2}f_{i-1} \\
&\quad - \frac{(1+s)(2+s)(3+s)}{6}f_{i-2}, \\
P_{2,3}(s) &= \frac{(1+s)(2+s)(3+s)}{6}f_{i+2} - \frac{(s)(2+s)(3+s)}{2}f_{i+1} + \frac{(s)(1+s)(3+s)}{2}f_i \\
&\quad - \frac{(s)(1+s)(2+s)}{6}f_{i-1}.
\end{aligned}$$

The limits of integration in Eq. (5),(7), and (9) are then replaced with -2 to -1, while the limits of integration in Eq. (6),(8), and (10) are -2 to 0. The first, second, and third integration are computed by using Maple.

The predictor-corrector approach is chosen in this study to increase the accuracy of the method by predicting the approximate solution then correcting it by using the method of a higher order. Accordingly, the predictor is chosen to be one order less than the corrector. The derivation of predictor method has a similar technique as the corrector method but with a difference degree of Lagrange interpolation polynomial, i.e. degree 2, which is as follows:

$$P(x) = \frac{(x-x_{i-1})(x-x_{i-2})}{(x_i-x_{i-1})(x_i-x_{i-2})}f_i + \frac{(x-x_i)(x-x_{i-2})}{(x_{i-1}-x_i)(x_{i-1}-x_{i-2})}f_{i-1} + \frac{(x-x_i)(x-x_{i-1})}{(x_{i-2}-x_i)(x_{i-2}-x_{i-1})}f_{i-2}.$$

The interpolation points chosen in the predictor for both points, y_{i+1} and y_{i+2} are the same. Finally, the predictor and corrector method for the two point block multistep method can be obtained as the following:

Predictor:

$$\begin{aligned}
y''_{i+1} &= y''_i + \frac{h}{12}(23f_i - 16f_{i-1} + 5f_{i-2}), \\
y'_{i+1} &= y'_i + hy''_i + \frac{h^2}{24}(19f_i - 10f_{i-1} + 3f_{i-2}), \\
y_{i+1} &= y_i + hy'_i + \frac{h^2}{2}y''_i + \frac{h^3}{240}(57f_i - 24f_{i-1} + 7f_{i-2}), \\
y''_{i+2} &= y''_i + \frac{h}{3}(19f_i - 20f_{i-1} + 7f_{i-2}), \\
y'_{i+2} &= y'_i + 2hy''_i + \frac{h^2}{3}(14f_i - 12f_{i-1} + 4f_{i-2}), \\
y_{i+2} &= y_i + 2hy'_i + 2h^2y''_i + \frac{h^3}{15}(39f_i - 28f_{i-1} + 9f_{i-2}).
\end{aligned} \tag{11}$$

Corrector:

$$
\begin{aligned}
y''_{i+1} &= y''_i + \frac{h}{24}\big(9f_{i+1} + 19f_i - 5f_{i-1} + f_{i-2}\big),\\
y'_{i+1} &= y'_i + hy''_i + \frac{h^2}{360}\big(38f_{i+1} + 171f_i - 36f_{i-1} + 7f_{i-2}\big),\\
y_{i+1} &= y_i + hy'_i + \frac{h^2}{2}y''_i + \frac{h^3}{720}\big(17f_{i+1} + 120f_i - 21f_{i-1} + 4f_{i-2}\big),\\
y''_{i+2} &= y''_i + \frac{h}{3}\big(f_{i+2} + 4f_{i+1} + f_i\big),\\
y'_{i+2} &= y'_i + 2hy''_i + \frac{h^2}{45}\big(2f_{i+2} + 54f_{i+1} + 36f_i - 2f_{i-1}\big),\\
y_{i+2} &= y_i + 2hy'_i + 2h^2y''_i + \frac{h^3}{45}\big(-f_{i+2} + 30f_{i+1} + 33f_i - 2f_{i-1}\big).
\end{aligned}
\tag{12}
$$

3 Analysis of Method

Order of Method:

The order of the method can be used to observe how well the method approximates the solutions. The general form for third order linear multistep method (LMM) is:

$$
\sum_{j=0}^{k}\alpha_j y(x_{i+j}) = h\sum_{j=0}^{k}\beta_j y'(x_{i+j}) + h^2\sum_{j=0}^{k}\gamma_j y''(x_{i+j}) + h^3\sum_{j=0}^{k}\sigma_j f\big(x_{i+j}, y_{i+j}, y'_{i+j}, y''_{i+j}\big). \tag{13}
$$

By writing the corrector method (12) following the LMM (13) above, the matrix form is obtained as below:

$$
\begin{bmatrix}
0&0&0&0&0&0\\
0&0&0&0&0&0\\
0&0&0&-1&1&0\\
0&0&0&0&0&0\\
0&0&0&0&0&0\\
0&0&0&-1&0&1
\end{bmatrix}
\begin{bmatrix} y_{i-3}\\ y_{i-2}\\ y_{i-1}\\ y_i\\ y_{i+1}\\ y_{i+2}\end{bmatrix}
= h
\begin{bmatrix}
0&0&0&0&0&0\\
0&0&0&1&-1&0\\
0&0&0&1&0&0\\
0&0&0&0&0&0\\
0&0&0&1&0&-1\\
0&0&0&2&0&0
\end{bmatrix}
\begin{bmatrix} y'_{i-3}\\ y'_{i-2}\\ y'_{i-1}\\ y'_i\\ y'_{i+1}\\ y'_{i+2}\end{bmatrix}
$$

$$
+h^2
\begin{bmatrix}
0&0&0&1&-1&0\\
0&0&0&1&0&0\\
0&0&0&\frac{1}{2}&0&0\\
0&0&0&1&0&-1\\
0&0&0&2&0&0\\
0&0&0&2&0&0
\end{bmatrix}
\begin{bmatrix} y''_{i-3}\\ y''_{i-2}\\ y''_{i-1}\\ y''_i\\ y''_{i+1}\\ y''_{i+2}\end{bmatrix}
+ h^3
\begin{bmatrix}
0&\frac{1}{24}&-\frac{5}{24}&\frac{19}{24}&\frac{9}{24}&0\\
0&\frac{7}{360}&-\frac{36}{360}&\frac{171}{360}&\frac{38}{360}&0\\
0&\frac{4}{720}&-\frac{21}{720}&\frac{120}{720}&\frac{17}{720}&0\\
0&0&0&\frac{1}{3}&\frac{4}{3}&\frac{1}{3}\\
0&0&-\frac{2}{45}&\frac{36}{45}&\frac{54}{45}&\frac{2}{45}\\
0&0&-\frac{2}{45}&\frac{33}{45}&\frac{30}{45}&-\frac{1}{45}
\end{bmatrix}
\begin{bmatrix} f_{i-3}\\ f_{i-2}\\ f_{i-1}\\ f_i\\ f_{i+1}\\ f_{i+2}\end{bmatrix}.
\tag{14}
$$

The order of the method can be determined by using the formula below:

$$\begin{aligned}
C_0 &= \alpha_0 + \alpha_1 + \alpha_2 + ... + \alpha_s,\\
C_1 &= \alpha_1 + 2\alpha_2 + 3\alpha_3 + ... + s\alpha_s - (\beta_0 + \beta_1 + \beta_2 + ... + \beta_s),\\
C_2 &= \frac{1}{2!}(\alpha_1 + 2^2\alpha_2 + 3^2\alpha_3 + ... + s^2\alpha_s) - (\beta_1 + 2\beta_2 + 3\beta_3 + ... + s\beta_s)\\
&\quad - (\gamma_0 + \gamma_1 + \gamma_2 + ... + \gamma_s),\\
C_3 &= \frac{1}{3!}(\alpha_1 + 2^3\alpha_2 + 3^3\alpha_3 + ... + s^3\alpha_s) - \frac{1}{2!}(\beta_1 + 2\beta_2 + 3\beta_3 + ... + s\beta_s)\\
&\quad - (\gamma_1 + 2\gamma_2 + 3\gamma_3 + ... + s\gamma_s) - (\sigma_0 + \sigma_1 + \sigma_2 + \sigma_3 + ... + \sigma_s),\\
&\vdots\\
C_r &= \frac{1}{r!}(\alpha_1 + 2^r\alpha_2 + 3^r\alpha_3 + ... + s^r\alpha_s) - \frac{1}{(r-1)!}(\beta_1 + 2^{r-1}\beta_2 + 3^{r-1}\beta_3 + ... + s^{r-1}\beta_s)\\
&\quad - \frac{1}{(r-2)!}(\gamma_1 + 2^{r-2}\gamma_2 + 3^{r-2}\gamma_3 + ... + s^{r-2}\gamma_s) - \frac{1}{(r-3)!}(\sigma_1 + 2^{r-3}\sigma_2 + 3^{r-3}\sigma_3\\
&\quad + ... + s^{r-3}\sigma_s),\\
&r = 4, 5, 6, \ldots .
\end{aligned}$$

Substituting the matrix form (14) into the formula above,

$$C_0 = C_1 = C_2 = C_3 = C_4 = C_5 = C_6 = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T,$$

and

$$C_7 = \begin{bmatrix}-\frac{19}{720} & -\frac{17}{1440} & -\frac{11}{3360} & -\frac{1}{90} & \frac{1}{90} & \frac{1}{70}\end{bmatrix}^T.$$

The LMM (13) is said to have order p if $C_{p+3} \neq 0$. The C_7 is the first non-zero coefficients, thus the proposed method is order 4. Hence, the proposed method in this paper shall be called as two point block multistep method order 4 (2PBM4).

Consistency of Method:

The linear difference operator for LMM (13) is expressed as below:

$$\begin{aligned}
L[y(x);h] &= \sum_{j=0}^{k}\alpha_j y(x_{i+j}) - h\sum_{j=0}^{k}\beta_j y'(x_{i+j}) - h^2\sum_{j=0}^{k}\gamma_j y''(x_{i+j})\\
&\quad - h^3\sum_{j=0}^{k}\sigma_j f\big(x_{i+j}, y_{i+j}, y'_{i+j}, y''_{i+j}\big),\\
L[y(x);h] &= C_{p+3}h^{p+3}y^{(p+3)}.
\end{aligned} \tag{15}$$

Definition 3.1. [4]: *The local truncation error (LTE) of the LMM (13) is represented as the $L[y(x);h]$ expression given by (15), where $y(x)$ is the theoretical solution to the problem.*

Therefore, the LTE of 2PBM4 is

$$C_7h^7y^{(7)} = h^7y^{(7)}\begin{bmatrix}-\frac{19}{720} & -\frac{17}{1440} & -\frac{11}{3360} & -\frac{1}{90} & \frac{1}{90} & \frac{1}{70}\end{bmatrix}^T.$$

The LMM (13) is said to be consistent when the step size, h is approaching zero, then the LTE is also approaching zero. Hence, we can deduce that 2PBM4 is consistent.

Zero Stability of Method:

Definition 3.2. [4]: *The LMM (13) is said to be zero stable if the roots of the first characteristic polynomial $\rho(R)$ do not have modulus greater than one and if the multiplicity of the root with modulus one is not greater than three.*

The first characteristic polynomial of 2PBM4 is as follows:

$$\rho(R) = \det\Big[\sum_{i=0}^{p} A_{(i)} R^{p-i}\Big] = \det\big[A_0 R^1 - A_1\big] = 0,$$

where

$$A_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad A_1 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Thus, we have,

$$\rho(R) = \det \begin{bmatrix} R & 0 & 0 & -1 & 0 & 0 \\ 0 & R & 0 & 0 & -1 & 0 \\ 0 & 0 & R & 0 & 0 & -1 \\ 0 & 0 & 0 & R-1 & 0 & 0 \\ 0 & 0 & 0 & 0 & R-1 & 0 \\ 0 & 0 & 0 & 0 & 0 & R-1 \end{bmatrix} = 0,$$

$$R^3(R-1)^3 = 0,$$

$$R = 0, 0, 0, 1, 1, 1.$$

The roots specify $|R_j| \leq 1$ and the multiplicity of $|R_j| = 1$ is not exceeding three, thus the 2PBM4 is said to be zero stable.

Convergence of Method:

Recalled the approximate solution in (12) and consider the exact solution of (1)-(2) as follows:

$$\begin{aligned}
Y''_{i+1} &= Y''_i + \frac{h}{24}\big(9F_{i+1} + 19F_i - 5F_{i-1} + F_{i-2}\big) + \frac{19}{720} h^5 Y^{(5)}(\zeta_i), \\
Y'_{i+1} &= Y'_i + hY''_i + \frac{h^2}{360}\big(38F_{i+1} + 171F_i - 36F_{i-1} + 7F_{i-2}\big) + \frac{17}{1440} h^5 Y^{(5)}(\zeta_i), \\
Y_{i+1} &= Y_i + hY'_i + \frac{h^2}{2}Y''_i + \frac{h^3}{720}\big(17F_{i+1} + 120F_i - 21F_{i-1} + 4F_{i-2}\big) + \frac{11}{3360} h^5 Y^{(5)}(\zeta_i), \\
Y''_{i+2} &= Y''_i + \frac{h}{3}\big(F_{i+2} + 4F_{i+1} + F_i\big) + \frac{1}{90} h^5 Y^{(5)}(\zeta_i), \\
Y'_{i+2} &= Y'_i + 2hY''_i + \frac{h^2}{45}\big(2F_{i+2} + 54F_{i+1} + 36F_i - 2F_{i-1}\big) + \frac{1}{90} h^5 Y^{(5)}(\zeta_i), \\
Y_{i+2} &= Y_i + 2hY'_i + 2h^2Y''_i + \frac{h^3}{45}\big(-F_{i+2} + 30F_{i+1} + 33F_i - 2F_{i-1}\big) + \frac{1}{70} h^5 Y^{(5)}(\zeta_i).
\end{aligned} \tag{16}$$

where $F = f\big(x, Y(x), Y'(x), Y(x-\tau), Y'(x-\tau)\big)$. Under the condition of the initial function, $y_i = \phi(x_i)$, then the error is

$$E_i = Y(x_i) - y_i = 0, \quad \text{for} \quad i = -m, -m+1, \ldots, 0.$$

Subtracting the approximate solution (12) from the exact solution (16) with the consideration of the error above then the Lipchitz condition is as below

$$\big|\big|f\big(x, Y(x), Y'(x)\big) - f\big(x, y(x), y'(x)\big)\big|\big| \leq L\big(||Y - y|| + ||Y' - y'||\big).$$

After rearranging the equation, we finally obtained

$$k_{i+1} - k_i \leq hL\sum_{j=0}^{3}\Big(\big|d_{i-2+j}\big| + \big|w_{i-2+j}\big|\Big) + \frac{19}{720}h^5Y^{(5)}(\zeta_i),$$

$$w_{i+1} - w_i \leq h\Bigg[k_i + hL\sum_{j=0}^{3}\Big(\big|d_{i-2+j}\big| + \big|w_{i-2+j}\big|\Big)\Bigg] + \frac{17}{1440}h^5Y^{(5)}(\zeta_i),$$

$$d_{i+1} - d_i \leq h\Bigg[w_i + \frac{h}{2}k_i + h^2L\sum_{j=0}^{3}\Big(\big|d_{i-2+j}\big| + \big|w_{i-2+j}\big|\Big)\Bigg] + \frac{11}{3360}h^5Y^{(5)}(\zeta_i),$$

$$k_{i+2} - k_i \leq hL\sum_{j=0}^{2}\Big(\big|d_{i+j}\big| + \big|w_{i+j}\big|\Big) + \frac{1}{90}h^5Y^{(5)}(\zeta_i),$$

$$w_{i+2} - w_i \leq h\Bigg[2k_i + hL\sum_{j=0}^{3}\Big(\big|d_{i-1+j}\big| + \big|w_{i-1+j}\big|\Big)\Bigg] + \frac{1}{90}h^5Y^{(5)}(\zeta_i),$$

$$d_{i+2} - d_i \leq h\Bigg[2w_i + 2hk_i + h^2L\sum_{j=0}^{3}\Big(\big|d_{i-1+j}\big| + \big|w_{i-1+j}\big|\Big)\Bigg] + \frac{1}{70}h^5Y^{(5)}(\zeta_i),$$

where $d_i = Y_i - y_i$, $w_i = Y_i' - y_i'$, $k_i = Y_i'' - y_i''$ and $L \geq 0$. The proposed method is said to be convergent when

$$\lim_{h\to 0} y_i = Y(x_i).$$

Thus, when h approaching zero, we obtained

$$\begin{aligned}
k_{i+1} - k_i \leq 0 &\Rightarrow y_{i+1}'' - y_i'' \leq Y_{i+1}'' - Y_i'',\\
w_{i+1} - w_i \leq 0 &\Rightarrow y_{i+1}' - y_i' \leq Y_{i+1}' - Y_i',\\
d_{i+1} - d_i \leq 0 &\Rightarrow y_{i+1} - y_i \leq Y_{i+1} - Y_i,\\
k_{i+2} - k_i \leq 0 &\Rightarrow y_{i+2}'' - y_i'' \leq Y_{i+2}'' - Y_i'',\\
w_{i+2} - w_i \leq 0 &\Rightarrow y_{i+2}' - y_i' \leq Y_{i+2}' - Y_i',\\
d_{i+2} - d_i \leq 0 &\Rightarrow y_{i+2} - y_i \leq Y_{i+2} - Y_i.
\end{aligned}$$

Hence, the approximate solution converges to the exact solution when h tends to zero.

4 Implementation of Method

4.1 Delay Differential Equations

Two types of delay are solved by 2PBM4 in this paper which are constant delay type (1)-(2) and pantograph delay type (3).

Constant Delay:

The delay term solutions, $y(x-\tau), y'(x-\tau)$, and $y''(x-\tau)$ will be approximated by using the initial function (2) and backward difference method if the delay argument, $(x-\tau)$ satisfies $(x-\tau) \in [a-\tau, a]$. The delay term solutions will take the values of the stored approximate solutions if the delay argument satisfies $(x-\tau) \in [a, b]$. This is possible considering that $\tau = mh$ where m is a positive integer and taking the delay as a step.

Pantograph Delay:

Pantograph delay is different than the constant delay since the delay term solutions do not require the initial function because the delay argument, (qx) always satisfies $(qx) \in [a, b]$. Hence, the delay term solutions have to be approximated by using the Lagrange interpolation polynomial. The order of the Lagrange polynomial is chosen to be one order higher than the proposed method, which is order 5 to assure the accuracy of the numerical results.

4.2 Boundary Value Problem

Boundary value problem (BVP) is changed to the initial value problem (IVP) by transforming the boundary conditions to the initial conditions, however, there is not enough information given on the initial conditions. Therefore, the shooting technique is applied to guess the additional initial condition.

Type I and Type II boundary conditions:

Consider that the ODEs (4) are interpreted in such a way that the dependent variable y relies on both x and variable t as the following IVP:

$$y'''(x,t) = f\big(x, y(x,t), y'(x,t), y''(x,t)\big) \tag{17}$$

with initial conditions:

$$y(a,t) = \alpha,\ y'(a,t) = \gamma,\ \text{and}\ y''(a,t) = t_1.$$

Partial differentiate on both sides of ODEs (17) with respect to t,

$$\frac{\partial y'''}{\partial t}(x,t) = \frac{\partial f}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial f}{\partial y'}\frac{\partial y'}{\partial t} + \frac{\partial f}{\partial y''}\frac{\partial y''}{\partial t}.$$

Supposed that $z(x,t) = \frac{\partial y}{\partial t}(x,t)$, then the second IVP is obtained as follows:

$$z'''(x,t) = \frac{\partial f}{\partial y}(x,y,y',y'')z(x,t) + \frac{\partial f}{\partial y'}(x,y,y',y'')z'(x,t) + \frac{\partial f}{\partial y''}(x,y,y',y'')z''(x,t) \tag{18}$$

with initial conditions:

$$z(a,t) = 0,\ z'(a,t) = 0,\ \text{and}\ z'(a,t) = 1.$$

The parameters $t = t_k$ is chosen such that

$$\lim_{k\to\infty} y(b,t_k) - \beta = 0.$$

The first initial guessing, t_1 is

$$t_1 = \frac{\beta - \alpha}{b - a}.$$

Then, to generate the next guessing t_k for $k = 2, 3, 4, \ldots$, the Newton's-like method is used as follows:

Type I:

$$t_k = w_{k-1} - \frac{y(b, w_{k-1}) - \beta}{z(b, w_{k-1})}$$

where

$$w_{k-1} = t_{k-1} - \frac{y(b, t_{k-1}) - \beta}{z(b, t_{k-1})}.$$

Type II:

$$t_k = w_{k-1} - \frac{y'(b, w_{k-1}) - \beta}{z'(b, w_{k-1})}$$

where

$$w_{k-1} = t_{k-1} - \frac{y'(b, t_{k-1}) - \beta}{z'(b, t_{k-1})}.$$

Due to the requirement of $y(b, w_{k-1}), z(b, w_{k-1})$ and their derivatives in the above formulas, both IVPs (17) and (18) have to be solved simultaneously to compute these solutions. The process will stop until $|y(b, t_{k-1}) - \beta| \leq \text{TOL}$ for Type I and $|y'(b, t_{k-1}) - \beta| \leq \text{TOL}$ for Type II where TOL is chosen to be smaller value.

Type III boundary conditions:

The procedure of shooting technique for Type III is quite similar as Type I and Type II, however, the difference is only in the initial conditions. The first IVP is

$$y'''(x, t) = f\big(x, y(x, t), y'(x, t), y''(x, t)\big), \tag{19}$$

with initial conditions:

$$y(a, t) = \alpha, \; y'(a, t) = t_1, \; \text{and } y''(a, t) = \gamma.$$

Meanwhile, the second IVP is

$$z'''(x, t) = \frac{\partial f}{\partial y}(x, y, y', y'')z(x, t) + \frac{\partial f}{\partial y'}(x, y, y', y'')z'(x, t) + \frac{\partial f}{\partial y''}(x, y, y', y'')z''(x, t), \tag{20}$$

with initial conditions:

$$z(a, t) = 0, \; z'(a, t) = 1, \; \text{and } z''(a, t) = 0.$$

By solving both IVPs (19) and (20) simultaneously will generate the solutions to be used in the Newton's-like method as below:

$$t_k = w_{k-1} - \frac{y(b, w_{k-1}) - \beta}{z(b, w_{k-1})},$$

where

$$w_{k-1} = t_{k-1} - \frac{y(b, t_{k-1}) - \beta}{z(b, t_{k-1})}.$$

The stopping criteria is $|y(b, t_{k-1}) - \beta| \leq \text{TOL}$.

5 Numerical Results

There are three numerical problems to be tested using the proposed method, 2PBM4, and compared with the previous multistep block method, Hoo4 in [7]. Each problem constitutes of Type I, Type II, and Type III boundary conditions.

Problem 1: Constant delay [Source: [1]]

$$y''' = -y(x) - y(x-0.3) + e^{-x+0.3}, \quad x \in [0,1],$$

$$y(x) = e^{-x}, \quad x \in [-1,0].$$

Boundary conditions:
Type I: $y(0)=1,\ y'(0)=-1,\ y(1)=e^{-1}$.
Type II: $y(0)=1,\ y'(0)=-1,\ y'(1)=-e^{-1}$.
Type III: $y(0)=1,\ y''(0)=1,\ y(1)=e^{-1}$.
Exact solution: $y(x)=e^{-x}, \quad x\in[0,1]$.

Problem 2: Constant delay [Source: [2]]

$$y''' = -2y'(x) - y\big(x-\frac{\pi}{2}\big), \quad x \in \big[0,\frac{\pi}{2}\big],$$

$$y(x) = \sin(x), \quad x \in \big[-\frac{\pi}{2},0\big].$$

Boundary conditions:
Type I: $y(0)=0,\ y'(0)=1,\ y\big(\frac{\pi}{2}\big)=1$.
Type II: $y(0)=0,\ y'(0)=1,\ y'\big(\frac{\pi}{2}\big)=0$.
Type III: $y(0)=0,\ y''(0)=0,\ y\big(\frac{\pi}{2}\big)=1$.
Exact solution: $y(x)=\sin(x), \quad x\in\big[0,\frac{\pi}{2}\big]$.

Problem 3: Pantograph delay [Source: [11]]

$$y''' = -1 + 2y^2\big(\frac{x}{2}\big), \quad x \in \big[0,1\big].$$

Boundary conditions:
Type I: $y(0)=0,\ y'(0)=1,\ y(1)=\sin(1)$.
Type II: $y(0)=0,\ y'(0)=1,\ y'(1)=\cos(1)$.
Type III: $y(0)=0,\ y''(0)=0,\ y(1)=\sin(1)$.
Exact solution: $y(x)=\sin(x), \quad x\in\big[0,1\big]$.

The following notations are used for Table 1–12:

h	:	Step size.
MAXE	:	Maximum absolute error.
AVE	:	Average errors.
ITN	:	Total number of guessing.
FCN	:	Total function calls in the last guessing iteration.
TS	:	Total steps.
t_k	:	The guessing value in last guessing iteration.
2PBM4	:	Two-Point Block Diagonally Implicit Method of order four in this study.
Hoo4	:	Two-Point Block Fully Implicit Method of order four in [7].
Time (s)	:	Execution time (CPU time) in seconds.

Table 1: The absolute errors of 2PBM4 for Problem 1.

x	Type I		Type II		Type III	
	h=0.1	h=0.01	h=0.1	h=0.01	h=0.1	h=0.01
0.0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.1	1.0849E-07	1.3811E-10	6.6184E-07	1.4719E-10	6.8395E-06	7.6944E-09
0.2	3.7626E-07	2.5525E-10	2.1047E-06	2.9155E-10	1.2894E-05	1.3745E-08
0.3	8.2335E-06	3.4050E-10	1.1206E-05	4.2216E-10	2.5033E-05	1.8130E-08
0.4	1.1868E-06	3.9351E-10	5.3773E-06	5.3859E-10	2.0654E-05	2.0833E-08
0.5	1.5167E-06	4.1380E-10	6.8943E-06	6.4024E-10	2.2017E-05	2.1831E-08
0.6	1.6428E-06	4.0073E-10	8.1709E-06	7.2629E-10	2.1513E-05	2.1091E-08
0.7	1.5580E-06	3.5357E-10	9.1919E-06	7.9564E-10	1.9098E-05	1.8576E-08
0.8	1.2583E-06	2.7154E-10	9.9425E-06	8.4704E-10	1.4733E-05	1.4247E-08
0.9	7.3927E-07	1.5390E-10	1.0405E-05	8.7904E-10	8.3778E-06	8.0660E-09
1.0	1.6653E-16	1.1102E-16	1.0564E-05	8.9007E-10	5.5511E-17	3.8858E-16

Table 2: The absolute errors of 2PBM4 for Probelm 2.

x	Type I		Type II		Type III	
	h=$\pi/20$	h=$\pi/200$	h=$\pi/20$	h=$\pi/200$	h=$\pi/20$	h=$\pi/200$
0.00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.16	5.9140E-06	1.3885E-09	7.8374E-06	6.4587E-09	1.2711E-04	1.3552E-07
0.31	1.7727E-05	2.6301E-09	2.5326E-05	2.8373E-08	2.3996E-04	2.4861E-07
0.47	2.7879E-05	3.5617E-09	4.4627E-05	6.4768E-08	3.2605E-04	3.3361E-07
0.63	3.4508E-05	4.1394E-09	6.3429E-05	1.1385E-07	3.7979E-04	3.8634E-07
0.79	3.7429E-05	4.3366E-09	8.0949E-05	1.7321E-07	3.9868E-04	4.0421E-07
0.94	3.6461E-05	4.1458E-09	9.6287E-05	2.3993E-07	3.8174E-04	3.8635E-07
1.10	3.1853E-05	3.5785E-09	1.0889E-04	3.1073E-07	3.3002E-04	3.3363E-07
1.26	2.3826E-05	2.6647E-09	1.1814E-04	3.8211E-07	2.4605E-04	2.4865E-07
1.41	1.3022E-05	1.4513E-09	1.2382E-04	4.5059E-07	1.3420E-04	1.3559E-07
1.57	2.2204E-16	2.2204E-16	1.2569E-04	5.1277E-07	0.0000E+00	2.2204E-16

Table 3: The absolute errors of 2PBM4 for Probelm 3.

x	Type I		Type II		Type III	
	h=0.1	h=0.01	h=0.1	h=0.01	h=0.1	h=0.01
0.0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.1	4.1133E-07	2.4250E-10	3.5232E-07	4.9470E-11	3.2995E-06	1.0692E-08
0.2	7.6795E-07	9.8406E-10	5.3191E-07	1.8383E-10	5.5551E-06	2.0546E-08
0.3	1.1412E-06	2.2202E-09	6.1007E-07	4.0758E-10	6.8388E-06	2.9570E-08
0.4	1.6350E-06	3.9513E-09	6.9073E-07	7.2073E-10	7.2563E-06	3.7769E-08
0.5	2.2234E-06	6.1778E-09	7.4780E-07	1.1233E-09	6.7849E-06	4.5157E-08
0.6	2.9058E-06	8.9010E-09	7.8030E-07	1.6155E-09	5.4283E-06	5.1754E-08
0.7	3.6799E-06	1.2123E-08	7.8558E-07	2.1976E-09	3.1903E-06	5.7591E-08
0.8	4.5495E-06	1.5847E-08	7.6647E-07	2.8702E-09	8.2143E-08	6.2707E-08
0.9	5.5138E-06	2.0079E-08	7.2111E-07	3.6343E-09	3.8887E-06	6.7155E-08
1.0	6.5776E-06	2.4826E-08	6.5240E-07	4.4911E-09	8.7085E-06	7.1002E-08

Table 4: The comparison results for Problem 1 (Type I).

h	METHOD	MAXE	AVE	FCN	TS	ITN	t_k	Time(s)
0.1	2PBM4	8.2335E-06	1.6620E-06	29	6	1	1.000025	0.001000
	Hoo4	8.9473E-06	4.2936E-06	29	6	1	1.000128	0.002000
0.01	2PBM4	4.1400E-10	2.7440E-10	201	51	1	1.000000	0.001333
	Hoo4	9.6969E-10	5.4010E-10	299	51	1	1.000000	0.001667
0.001	2PBM4	4.2633E-14	2.8460E-14	2001	501	1	1.000000	0.003000
	Hoo4	9.9143E-14	5.6597E-14	2999	501	1	1.000000	0.003333

Table 5: The comparison results for Problem 1 (Type II).

h	METHOD	MAXE	AVE	FCN	TS	ITN	t_k	Time(s)
0.1	2PBM4	1.1206E-05	7.4519E-06	21	6	1	1.000156	0.001333
	Hoo4	1.3537E-05	8.2356E-06	29	6	1	1.000149	0.001667
0.01	2PBM4	8.9007E-10	5.7862E-10	201	51	1	1.000000	0.001000
	Hoo4	1.2788E-09	8.6441E-10	299	51	1	1.000000	0.001333
0.001	2PBM4	9.0816E-14	5.8772E-14	2001	501	1	1.000000	0.004000
	Hoo4	1.3012E-13	8.8506E-14	2999	501	1	1.000000	0.004333

Table 6: The comparison results for Problem 1 (Type III).

h	METHOD	MAXE	AVE	FCN	TS	ITN	t_k	Time(s)
0.1	2PBM4	2.5033E-05	1.5116E-05	21	6	1	-0.999930	0.001000
	Hoo4	2.5850E-05	1.5498E-05	29	6	1	-0.999933	0.001333
0.01	2PBM4	2.1836E-08	1.4566E-08	201	51	1	-0.999999	0.001333
	Hoo4	2.2376E-08	1.4847E-08	299	51	1	-0.999999	0.002000
0.001	2PBM4	2.2037E-11	1.4703E-11	2001	501	1	-1.000000	0.002667
	Hoo4	2.2092E-11	1.4732E-11	2999	501	1	-1.000000	0.004667

Table 7: The comparison results for Problem 2 (Type I).

h	METHOD	MAXE	AVE	FCN	TS	ITN	t_k	Time(s)
$\frac{\pi}{20}$	2PBM4	3.7429E-05	2.2862E-05	21	6	1	-0.000579	0.001333
	Hoo4	6.6127E-05	2.2511E-05	29	6	1	-0.000421	0.001667
$\frac{\pi}{200}$	2PBM4	4.3366E-09	2.8111E-09	201	51	1	0.000000	0.001000
	Hoo4	5.7869E-08	2.9677E-08	299	51	1	0.000000	0.001333
$\frac{\pi}{2000}$	2PBM4	4.4109E-13	2.8759E-13	2001	501	1	0.000000	0.002000
	Hoo4	5.7761E-11	3.1120E-11	2999	501	1	0.000000	0.002333

Table 8: The comparison results for Problem 2 (Type II).

h	METHOD	MAXE	AVE	FCN	TS	ITN	t_k	Time(s)
$\frac{\pi}{20}$	2PBM4	1.2569E-04	7.9500E-05	21	6	1	-0.000736	0.001000
	Hoo4	2.5492E-04	1.1501E-04	29	6	1	-0.000154	0.001000
$\frac{\pi}{200}$	2PBM4	5.1277E-07	2.0472E-07	201	51	1	0.000000	0.001000
	Hoo4	3.5172E-07	1.7000E-07	299	51	1	0.000000	0.002333
$\frac{\pi}{2000}$	2PBM4	5.1803E-10	2.0703E-10	2001	501	1	0.000000	0.002333
	Hoo4	3.4919E-10	1.7063E-10	2999	501	1	0.000000	0.004000

Table 9: The comparison results for Problem 2 (Type III).

h	METHOD	MAXE	AVE	FCN	TS	ITN	t_k	Time(s)
$\frac{\pi}{20}$	2PBM4	3.9868E-04	2.5636E-04	21	6	1	0.999176	0.000667
	Hoo4	2.5808E-04	1.5173E-04	29	6	1	0.999400	0.001000
$\frac{\pi}{200}$	2PBM4	4.0421E-07	2.6364E-07	201	51	1	0.999999	0.001333
	Hoo4	2.3209E-07	1.4669E-07	299	51	1	0.999999	0.001667
$\frac{\pi}{2000}$	2PBM4	4.0441E-10	2.6381E-10	2001	501	1	1.000000	0.002667
	Hoo4	2.2927E-10	1.4621E-10	2999	501	1	1.000000	0.003000

Table 10: The comparison results for Problem 3 (Type I).

h	METHOD	MAXE	AVE	FCN	TS	ITN	t_k	Time(s)
0.1	2PBM4	5.5138E-06	2.2828E-06	21	6	1	0.000099	0.001000
	Hoo4	1.1769E-05	3.9743E-06	29	6	1	0.000099	0.001333
0.01	2PBM4	2.4328E-08	8.1283E-09	201	51	2	0.000000	0.001333
	Hoo4	2.3835E-08	7.9397E-09	299	51	2	0.000000	0.001333
0.001	2PBM4	9.4307E-11	3.1378E-11	2001	501	2	0.000000	0.003667
	Hoo4	9.4377E-11	3.1398E-11	2999	501	2	0.000000	0.004000

Table 11: The comparison results for Problem 3 (Type II).

h	METHOD	MAXE	AVE	FCN	TS	ITN	t_k	Time(s)
0.1	2PBM4	7.8558E-07	5.9863E-07	21	6	2	0.000087	0.000333
	Hoo4	6.9087E-06	2.2674E-06	29	6	2	0.000087	0.002333
0.01	2PBM4	4.4012E-09	1.4751E-09	201	51	2	0.000000	0.001000
	Hoo4	5.0858E-09	1.6606E-09	299	51	2	0.000000	0.001333
0.001	2PBM4	5.0104E-11	1.6671E-11	2001	501	3	0.000000	0.010000
	Hoo4	5.0035E-11	1.6652E-11	2999	501	3	0.000000	0.011333

Table 12: The comparison results for Problem 3 (Type III).

h	METHOD	MAXE	AVE	FCN	TS	ITN	t_k	Time(s)
0.1	2PBM4	7.2563E-06	4.2324E-06	21	6	1	1.000033	0.001000
	Hoo4	1.0383E-05	5.1343E-06	29	6	1	1.000034	0.001333
0.01	2PBM4	7.0642E-08	4.1552E-08	201	51	2	1.000000	0.001667
	Hoo4	7.1336E-08	4.1739E-08	299	51	2	1.000000	0.002000
0.001	2PBM4	5.7232E-10	2.7633E-10	2001	501	2	1.000000	0.003333
	Hoo4	5.7225E-10	2.7631E-10	2999	501	2	1.000000	0.004000

6 Discussion

Tables 1–3 display the absolute errors of each iteration for each boundary condition types as step size, h decreases. We can examine that as h tends to zero, the absolute error is smaller, hence we can conclude that the approximate solution of the 2PBM4 converges to the exact solution.

Tables 4–12 portray that the proposed method, 2PBM4 is either in agreement or slightly better than Hoo4 in terms of accuracy for Type I, II, and III as the step size, h is decreasing. This is due to the slight difference in deriving these two methods where 2PBM4 is derived in a diagonally implicit manner while Hoo4 is in a fully implicit manner that also gives reason to the difference in total function calls (FCN). Method 2PBM4 has fewer FCN compared to Hoo4 for all the problems in Tables 4–12. These fewer FCN will give a slight effect to the computation time as the programs of 2PBM4 in C programming runs either at the same speed or slightly faster than Hoo4 by using the same CPU laptop for all the problems in Tables 4–12.

Meanwhile, the total iteration steps (TS) for both methods are the same for all the results in Tables 4–12 because both methods compute the two approximate solutions simultaneously in the two point block approach for every iteration. Furthermore, the total number of guessing for the initial condition (ITN) is only one for both methods when solving linear Problem 1 and Problem 2 as the results in Tables 4–9. In Tables 10–12, we could observed that the ITN is more than one because Problem 3 is a non-linear problem. In non-linear problem, the Newton's-like method will be guessing more to obtain the most accurate additional initial conditions such as shown in Table 11 when h=0.001. The guessing value in the last iteration (t_k) for 2PBM4 is comparable to the Hoo4 for all the problems in Tables 4–12.

7 Conclusions

The proposed method, 2PBM4 has slightly better accuracy and faster computation time when compared to method Hoo4. The fewer total function calls of 2PBM4 than Hoo4 also give an advantage to the proposed method. To conclude, the 2PBM4 method proposed in this study is capable of solving directly the constant and pantograph delay of the third order DDEs with three different types of boundary conditions by obtaining better accuracy and less timing.

Acknowledgement The authors are grateful that this research was funded under the Putra Grant (GP-IPS/2018/9625200) from Universiti Putra Malaysia.

Conflicts of Interest The authors declare no conflict of interest.

References

[1] D. J. Evans & K. R. Raslan (2005). The Adomian decomposition method for solving delay differential equation. *International Journal of Computer Mathematics*, *82*(1), 49–54.

[2] S. R. Grace (2015). Oscillation criteria for third order nonlinear delay differential equations with damping. *Opuscula Mathematica*, *35*(4), 485–497.

[3] N. I. N. Ismail, Z. A. Majid & N. Senu (2019). Explicit multistep block method for solving neutral delay differential equation. *ASM Science Journal* (*IQRAC2018*), *12*(1), 24–32.

[4] J. D. Lambert (1973). *Computational methods in ordinary differential equations*. Wiley, New York.

[5] K. D. Nevers & K. Schmitt (1971). An application of the shooting method to boundary value problems for second order delay equations. *Journal of Mathematical Analysis and Applications*, *36*(3), 588–597.

[6] A. Saadatmandi & M. Dehghan (2009). Variational iteration method for solving a generalized pantograph equation. *Computers & Mathematics with Applications*, *58*(11-12), 2190–2196.

[7] H. Y. Seong (2016). *Multistep Blocks Methods For Solving Higher Order Delay Differential Equations*. PhD thesis, Universiti Putra Malaysia, Selangor, Malaysia.

[8] H. Y. Seong & Z. A. Majid (2017). Solving second order delay differential equations using direct two-point block method. *Ain Shams Engineering Journal*, *8*(1), 59–66.

[9] H. Y. Seong, Z. A. Majid & F. Ismail (2013). Solving second-order delay differential equations by direct Adams-Moulton method. *Mathematical Problems in Engineering*, *2013*(Article ID: 261240, 7 pages). https://doi.org/10.1155/2013/261240.

[10] M. Sezer & A. Akyüz-Daşcıoglu (2007). A Taylor method for numerical solution of generalized pantograph equations with linear functional argument. *Journal of Computational and Applied Mathematics*, *200*(1), 217–225.

[11] O. A. Taiwo & O. S. Odetunde (2010). On the numerical approximation of delay differential equations by a decomposition method. *Asian Journal of Mathematics and Statistics*, *3*(4), 237–243.

[12] S. K. Vanani, J. S. Hafshejani, F. Soleymani & M. Khan (2011). On the numerical solution of generalized pantograph equation. *World Applied Sciences Journal*, *13*(12), 2531–2535.