Menemui Matematik
(Discovering Mathematics)

journal homepage: https://persama.org.my/dismath/home

Predictor-Corrector Scheme with Off-step for solving Second-order Delay Differential Equations

Mohamat Aidil Mohamat Johari[1*] and Zanariah Abdul Majid[1]
[1]*Department of Mathematics, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor*

[1]mamj@upm.edu.my
*Corresponding author

Received: 21 August 2025
Accepted: 18 December 2025

ABSTRACT

In this study, the direct method of the Adams-Moulton three-step method with the off-step method was developed to solve the initial value problem (IVP) for the second-order delay differential equations (DDEs) directly. This method will be used to solve the second-order DDEs directly without reducing them to the first-order DDEs. Approach: Lagrange interpolation polynomial was applied in the derivation of the proposed method. Results: Numerical results confirmed that the method gave better accuracy and converged faster compared to the existing method. Conclusions: The proposed direct method is suitable for solving second-order DDEs.

Keywords: Delay differential equations; Direct method; Off-step method; Numerical simulation

INTRODUCTION

In recent years, significant attention has been devoted to the development of mathematical models involving delay differential equations (DDEs) across various fields of science and engineering. A DDE is characterized by the fact that the derivative of an unknown function at a given time depends on the values of the function at previous times, distinguishing it fundamentally from ordinary differential equations (ODEs). The use of DDE-based mathematical models has grown rapidly due to their ability to describe a wide range of real-world phenomena with greater accuracy. Such models are applied in diverse areas, including biology, economics, and mechanics. For instance, in biology, the well-known Mackey–Glass equation models the density of certain blood cells, where the delay represents the time between the initiation of cell production in the bone marrow and the release of mature cells into the bloodstream.

In this research, we will consider the general form of the Initial Value problem (IVPs) for the second-order delay differential equations (DDEs) as follows:

$$y'' = f(t, y(t), y(t-\tau)), a \le t \le b, \tau > 0 \tag{1}$$
$$y'(a) = \Omega$$
$$y(t) = \phi(t), \alpha \le t \le a, 0 \le \tau \le |a-\alpha|.$$

where $\phi(t)$ is an initial function and τ is a delay term.

The direct Adams-Moulton methods were studied by several researchers and have been proven to solve first-order, second-order, and higher-order ODE problems. In 2006, Majid and Suleiman proposed a direct integration implicit variable step method to solve higher order systems of ODEs. Majid et al (2009) solved second-order ODEs using the direct implicit block method. In 2022, Johari and Majid. solved second-order differential equations using Adam Bashforth-Moulton directly.

Recently, considerable research attention has been directed toward the numerical solution of delay differential equations (DDEs), particularly through the use of one-step methods. For instance, Enright and Hu (1995) addressed DDE problems employing the Interpolating Runge–Kutta method, while Karoui and Vaillancourt (1995) proposed a numerical technique specifically designed for vanishing-lag DDEs. In a similar vein, Suleiman and Ismail (2001) utilized the Runge–Kutta method with componentwise partitioning to obtain numerical solutions of DDEs. Subsequently, research efforts expanded to include multistep methods through the adaptation of block techniques. Notably, Rasdi and Majid (2015)implemented a two-point block method, originally introduced by Majid and Suleiman (2011), incorporating variable step sizes and six-point Lagrange interpolation to approximate the delayed terms. Furthermore, Ishak and Ahmad (2011) applied a predictor–corrector strategy based on Lagrange and Hermite interpolations to solve DDEs. In 2013, Hoo et al. developed the Direct Adams–Moulton method for second-order DDEs, while Sabir et al. (2020) investigated the use of both Adams and explicit Runge–Kutta schemes for similar problems. More recently, Ismail and Majid (2024) introduced a two-point, two off-step point block multistep method aimed at solving constant-type neutral delay Volterra integro-differential equations.

This paper aims to present a predictor–corrector one-step method, formulated in a simplified form of the Adams–Moulton approach, for solving equation (1) with a variable step size. The numerical results obtained are subsequently compared with those produced by Haar Wavelet and the Laplace transform method.

THE DIRECT METHOD

Formulation of the method

Most numerical techniques developed for ordinary differential equations (ODEs) can also be applied to delay differential equations (DDEs). The derivation of the one-point direct Adams–Moulton method is presented below. The point, y_{n+1} at t_{n+1} can be obtained by integrating equation (1) over the interval $[t_n, t_{n+1}]$.

By integrating once, we have:

$$\int_{t_n}^{t_{n+1}} y''(t)dt = \int_{t_n}^{t_{n+1}} f(t, y, y')dt.$$

Therefore

$$y'(t_{n+1}) - y'(t_n) = \int_{t_n}^{t_{n+1}} f(t, y, y')dt. \tag{2}$$

By integrating twice, we arrive at:

$$\int_{t_n}^{t_{n+1}} \int_{t_n}^{t} y''(t)dtdt = \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} f(t, y, y')dtdt.$$

Therefore

$$y(t_{n+1}) - y(t_n) - hy'(t_n) = \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} f(t, y, y') dt dt. \quad (3)$$

By replacing $f(t, y, y')$ in equations (2) and (3) with the polynomial interpolation with four interpolation points $t_{n-2}, t_{n-1}, t_{n+1/2}$ and t_{n+1}, we have

$$\begin{aligned} y'(t_{n+1}) - y'(t_n) &= \int_{t_n}^{t_{n+1}} [\frac{(t - t_{n+1/2})(t - t_{n-1})(t - t_{n-2})}{(t_{n+1} - t_{n+1/2})(t_{n+1} - t_{n-1})(t_{n+1} - t_{n-2})} f_{n+1} \\ &+ \frac{(t - t_{n+1})(t - t_{n-1})(t - t_{n-2})}{(t_{n+1/2} - t_{n+1})(t_{n+1/2} - t_{n-1})(t_{n+1/2} - t_{n-2})} f_{n+1/2} \\ &+ \frac{(t - t_{n+1})(t - t_{n+1/2})(t - t_{n-2})}{(t_{n-1} - t_{n+1})(t_{n-1} - t_{n+1/2})(t_{n-1} - t_{n-2})} f_{n-1} \\ &+ \frac{(t - t_{n+1})(t - t_{n+1/2})(t - t_{n-2})}{(t_{n-2} - t_{n+1})(t_{n-2} - t_{n+1/2})(t_{n-2} - t_{n-1})} f_{n-2}] dt, \end{aligned} \quad (4)$$

$$\begin{aligned} y(t_{n+1}) - y(t_n) - hy'(t_n) &= \int_{t_n}^{t_{n+1}} (t_{n+1} - t) [\frac{(t - t_{n+\frac{1}{2}})(t - t_{n-1})(t - t_{n-2})}{(t_{n+1} - t_{n+\frac{1}{2}})(t_{n+1} - t_{n-1})(t_{n+1} - t_{n-2})} f_{n+1} \\ &+ \frac{(t - t_{n+1})(t - t_{n-1})(t - t_{n-2})}{(t_{n+\frac{1}{2}} - t_{n+1})(t_{n+\frac{1}{2}} - t_{n-1})(t_{n+\frac{1}{2}} - t_{n-2})} f_{n+\frac{1}{2}} \\ &+ \frac{(t - t_{n+1})(t - t_{n+\frac{1}{2}})(t - t_{n-2})}{(t_{n-1} - t_{n+1})(t_{n-1} - t_{n+\frac{1}{2}})(t_{n-1} - t_{n-2})} f_{n-1} \\ &+ \frac{(t - t_{n+1})(t - t_{n+\frac{1}{2}})(t - t_{n-2})}{(t_{n-2} - t_{n+1})(t_{n-2} - t_{n+\frac{1}{2}})(t_{n-2} - t_{n-1})} f_{n-2}] dt. \end{aligned} \quad (5)$$

By substituting $s = \frac{t - t_{n+1}}{h}$ h and $dt = hds$, the corrector formulae can be derived by integrating (4) and (5).

Corrector formulae:

$$\begin{aligned} y'_{n+1} &= y'_n + \frac{h}{90}\left(10f_{n+1} + 76f_{n+1/2} + 5f_{n-1} - f_{n-2}\right) \\ y_{n+1} &= y_n + hy'_n + \frac{h^2}{1800}\left(95f_{n+1} - 912f_{n+\frac{1}{2}} - 105f_{n-1} + 22f_{n-2}\right). \end{aligned} \quad (6)$$

The predictor formulae was derived from Majid et al. (2011).

Predictor formulae:

$$y'_{n+1} = y'_n + \frac{h}{12}(23f_n - 16f_{n-1} + 5f_{n-2})$$
$$y_{n+1} = y_n + hy'_n + \frac{h^2}{24}(19f_n - 10f_{n-1} + 3f_{n-2}).$$

The off-step formulae are as follows,

$$y'_{n+1/2} = y'_n + \frac{h}{24}(17f_n - 7f_{n-1} + 2f_{n-2})$$
$$y_{n+1/2} = y_n + \frac{1}{2}hy'_n + \frac{h^2}{384}(-61f_n + 18f_{n-1} - 5f_{n-2}).$$

STABILITY ANALYSIS

Order of the Method

To check the order of the method, the corrector formula (6) is written in a matrix form of the linear multistep method (LMM) for second order as follows:

$$\alpha Y_i = h\beta Y_i' + h^2\gamma F_i. \tag{7}$$

From Eq. (6), we have

$$0 = -y'_{n+1} + y'_n + \frac{h}{90}\left(10f_{n+1} + 76f_{n+1/2} + 5f_{n-1} - f_{n-2}\right)$$
$$y_{n+1} - y_n = hy'_n + \frac{h^2}{1800}\left(95f_{n+1} - 912f_{n+\frac{1}{2}} - 105f_{n-1} + 22f_{n-2}\right)$$

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \end{pmatrix}\begin{pmatrix} y_{n-2} \\ y_{n-1} \\ y_n \\ y_{n+1} \\ y_{n+2} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}\begin{pmatrix} y_{n-2} \\ y_{n-1} \\ y_n \\ y_{n+1} \\ y_{n+2} \end{pmatrix}$$
$$+h^2\begin{pmatrix} -\frac{1}{90} & \frac{5}{90} & 0 & \frac{10}{90} & 0 \\ -\frac{22}{1800} & -\frac{105}{1800} & 0 & \frac{95}{1800} & 0 \end{pmatrix}\begin{pmatrix} f_{n-2} \\ f_{n-1} \\ f_n \\ f_{n+1} \\ f_{n+2} \end{pmatrix} + h^2\begin{pmatrix} \frac{76}{90} \\ -\frac{912}{1800} \end{pmatrix}\left(f_{n+1/2}\right).$$

Then we will obtain

$$\alpha_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \alpha_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \alpha_2 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \alpha_3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\beta_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \beta_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \beta_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \beta_3 = \begin{pmatrix} -1 \\ 0 \end{pmatrix} \tag{8}$$

$$\gamma_0 = \begin{pmatrix} -\frac{1}{90} \\ -\frac{22}{1800} \end{pmatrix}, \gamma_1 = \begin{pmatrix} \frac{5}{90} \\ -\frac{105}{1800} \end{pmatrix}, \gamma_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \gamma_3 = \begin{pmatrix} \frac{10}{90} \\ \frac{95}{1800} \end{pmatrix}, \gamma_4 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \gamma_5 = \begin{pmatrix} \frac{76}{90} \\ -\frac{912}{1800} \end{pmatrix}$$

The formula is defined as

$$C_0 = \alpha_0 + \alpha_1 + \alpha_2 + \cdots + a_k$$

$$C_1 = \alpha_1 + 2\alpha_2 + \cdots + k\alpha_k - (\beta_0 + \beta_1 + \cdots + \beta_k)$$

$$C_2 = \frac{1}{2!}(\alpha_1 + 2^2\alpha_2 + \cdots + k^2\alpha_k) - (\beta_1 + 2\beta_2 + \cdots + k\beta_k) - (\gamma_0 + \gamma_1 + \cdots + \gamma_k) \qquad (9)$$

$$\cdots$$

$$C_r = \sum_{j=0}^{k}\left(\frac{j^r}{r!}\alpha_j - \frac{j^{r-1}}{(r-1)!}\beta_j - \frac{j^{r-2}}{(r-2)!}\gamma_j\right) \text{ where } r = 3,4,5, \ldots.$$

Substituting value (8) into (9).

$$C_0 = \frac{1}{0!}(\alpha_0 + \alpha_1 + \alpha_2 + \alpha_3) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$C_1 = \frac{1}{1!}(\alpha_1 + 2^1\alpha_2 + 3^1\alpha_3) - \frac{1}{0!}(\beta_0 + \beta_1 + \beta_2 + \beta_3) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$C_2 = \frac{1}{2!}(\alpha_1 + 2^2\alpha_2 + 3^2\alpha_3) - \frac{1}{1!}(\beta_1 + 2\beta_2 + 3\beta_3) - \frac{1}{0!}\left(\gamma_0 + \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 + \frac{5^0\gamma_5}{2^0}\right) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$C_3 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, C_4 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, C_5 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$C_6 = \frac{1}{6!}(\alpha_1 + 2^6\alpha_2 + 3^6\alpha_3) - \frac{1}{5!}(\beta_1 + 2^5\beta_2 + 3^5\beta_3) - \frac{1}{4!}\left(\gamma_1 + 2^4\gamma_2 + 3^4\gamma_3 + 4^4\gamma_4 + \frac{5^4\gamma_5}{2^4}\right) = \begin{pmatrix} \frac{19}{2880} \\ \frac{23}{2880} \end{pmatrix}.$$

Lambert (1973) established that the order of a system is $n$if $C_0 = C_1 = \cdots = C_n = C_{n+1} = 0$and $C_{n+2} \neq 0$. Since $C_6 \neq 0$, the proposed method is determined to be of order 4. Accordingly, we refer to this scheme as the **Direct Adams–Moulton One-Off Step (DAM1OS)** method.

Consistency of Method

Assume that Z_i, Z_i', and Z_i'' below represent the matrices of theoretical solutions for DDE.

$$Zi = \begin{pmatrix} y(t_{i-2}) \\ y(t_{i-1}) \\ y(t_i) \\ y(t_{i+1}) \\ y(t_{i+2}) \end{pmatrix},$$

$$Z'i = \begin{pmatrix} y'(t_{i-2}) \\ y'(t_{i-1}) \\ y'(t_i) \\ y'(t_{i+1}) \\ y'(t_{i+2}) \end{pmatrix},$$

and

$$Z''i = \begin{pmatrix} f\big(t_{i-2}, y(t_{i-2}), y'(t_{i-2})\big) \\ f\big(t_{i-1}, y(t_{i-1}), y'(t_{i-1})\big) \\ f\big(t_i, y(t_i), y'(t_i)\big) \\ f\big(t_{i+1}, y(t_{i+1}), y'(t_{i+1})\big) \\ f\big(t_{i+2}, y(t_{i+2}), y'(t_{i+2})\big) \end{pmatrix}.$$

According to Fatunla (1991), the local truncation error (LTE) of the LMM (7) is introduced as

$$E_i = \alpha Z_i - h\beta Z_i' - h^2\gamma Z_i'',$$
$$||E_i|| = C_{n+2}h^{n+2} + O(h^{n+3}),$$

where ||.|| is the maximum norm. The maximum norm of LTE for the DAM1OS is

$$||Ei|| = h^6 \begin{pmatrix} \frac{19}{2880} \\ \frac{23}{2880} \end{pmatrix}.$$

The proposed DAM1OS method is consistent because, as the step size h approaches zero, the error term $\| E_i \|$ also approaches zero. Therefore, the DAM1OS method satisfies the consistency condition.

Zero Stability of Method

Zero stability is a necessary condition to ensure the stability of the method for a given step size h_0. Let the first characteristic polynomial $\rho(V)$be defined as

$$\rho(V) = \det\,[A_0V - A_1] = 0.$$

If all roots $V_j$of this polynomial satisfy $| V_j | \leq 1$, and any roots on the unit circle ($| V_j | = 1$)have multiplicity not exceeding two, then the DAM1OS method is deemed zero stable.

Based on the corrector formula equation (6), $A_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. Then

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

$$(V-1)^2 = 0, \qquad V = 1,1.$$

Since $|V_j| \leq 1$, DAM1OS is zero stable

Convergence of Method

Since the proposed method fulfils the criteria of consistency and zero stability, Dahlquist's convergence theorem guarantees that the DAM1OS method converges to the exact solution.

Stability of the Method

In this section, we will discuss the stability of the proposed method when applying the test equation

$$y'' = f = \lambda y(t) + \beta y(t - \tau)$$

into the proposed method. The method can be described in the following matrix form.

$$A_0 Y_{N+1} = A_1 Y_N + h \sum_{i=0}^{2} B_{i+1} F_{N-i} + h^2 \sum_{i=0}^{2} C_{i+1} F_{N+i}.$$

We have

$$A_0 = \begin{pmatrix} 1 & -\frac{10}{90} h\lambda \\ 0 & 1 + \frac{95}{1800} h^2 \lambda \end{pmatrix},\ A_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

$$B_1 = \begin{pmatrix} 0 & \frac{76}{90}\lambda \\ 0 & 0 \end{pmatrix}, B_2 = \begin{pmatrix} 0 & \frac{10}{90}\mu \\ 1 & 0 \end{pmatrix}, B_3 = \begin{pmatrix} 0 & \frac{76}{90}\mu \\ 0 & 0 \end{pmatrix}, B_4 = \begin{pmatrix} 0 & \frac{5}{90}\lambda \\ 0 & 0 \end{pmatrix},\ B_5 = \begin{pmatrix} 0 & \frac{5}{90}\mu - \frac{1}{90}\lambda \\ 0 & 0 \end{pmatrix}, B_6 = \begin{pmatrix} 0 & -\frac{1}{90}\mu \\ 0 & 0 \end{pmatrix},$$

$$C_1 = \begin{pmatrix} 0 & 0 \\ 0 & \frac{912}{1800}\lambda \end{pmatrix}, C_2 = \begin{pmatrix} 0 & 0 \\ 0 & -\frac{95}{1800}\mu \end{pmatrix}, C_3 = \begin{pmatrix} 0 & 0 \\ 0 & \frac{912}{1800}\mu \end{pmatrix}, C_4 = \begin{pmatrix} 0 & 0 \\ 0 & \frac{105}{1800}\lambda \end{pmatrix}, C_5 = \begin{pmatrix} 0 & 0 \\ 0 & \frac{105}{1800}\mu - \frac{22}{1800}\lambda \end{pmatrix}, C_6 = \begin{pmatrix} 0 & 0 \\ 0 & -\frac{22}{1800}\mu \end{pmatrix}.$$

Solving the determinant of

$$t^6(A_0) - t^5(A_1 + hB_1 + h^2C_1) - t^4(hB_2 + h^2C_2) - t^3(hB_3 + h^2C_3) - t^2(hB_4 + h^2C_4) - t(hB_5 + h^2C_5) - (hB_6 + h^2C_6) = 0.$$

By substituting $Y = h^2\beta$ and $X = h\lambda$, the stability polynomial is obtained.

$$t^4 + t^6 - 2t^5 - \frac{1207}{1800}t^5 Y - \frac{1}{9}t^6 X - \frac{713}{1800}t^4 Y + \frac{19}{360}t^6 Y - \frac{11}{15}t^5 X + \frac{71}{90}t^4 X + \frac{3}{200}t^3 Y - \frac{1}{900}t^2 Y + \frac{1}{15}t^3 X - \frac{1}{90}t^2 X = 0.$$

The boundary of the stability region in the X-Y plane is determined by substituting the values $t = 0, -1$, and $e^{i\theta}$ with $0 \leq \theta \leq 2\pi$ into the stability polynomial. Figure 1 illustrates the stability region of the direct method, where the bounded shaded area represents the stable region.

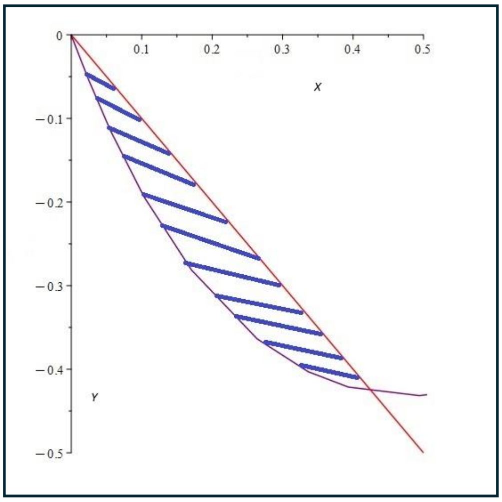

Figure 1: Stability region of DAM1OS

IMPLEMENTATION OF THE METHOD

The DAM1OS method for solving second-order delay differential equations (DDEs) is implemented within a predictor-corrector (PECE) framework. Since the proposed method is a multistep technique, it requires a set of preceding values before application. Specifically, three initial values y_0, y_1, and y_2, are necessary and can be obtained using a one-step method, such as Euler's method, prior to employing DAM1OS. Once these initial values are established, the DAM1OS method functions as the corrector, while the predictor is based on the equation presented in Majid et al. (2011).

For problems with constant delay, when $(t-\tau) \leq a$, the delay terms $y(t-\tau)$ and $y'(t-\tau)$ are evaluated using the initial function $\varphi(t)$. Conversely, when $(t-\tau) \geq a$, the delay terms depend on the location of $(t-\tau)$. Due to the fixed step size assumption $\tau = mh$ for $m = 1,2,3, \ldots$, these delayed values can be retrieved from previously stored solution values.

In this project, the DAM1OS algorithm was implemented in the C programming language with a predetermined step size selection.

Algorithm of the DAM1OS method

Step 1: Initialize the starting value k, ending value l, step size h, the given initial value, and the initial function $\varphi(t)$.
Step 2: For $n = 0,1,2$, set $t_{n+1} = k + nh$, compute the function $f_n$and the delay term d_n. Then, approximate $y'_{n+1}$and y_{n+1}using the Euler method.
Step 3: For $n \geq 3$, while $t_n < l$, proceed with Steps 4 through 6.
Step 4: Update $t_{n+1} = t_n + h$, compute $f_n$and the delay term d_n, and calculate the predicted values of $y'_{n+1}$and y_{n+1}using the predictor formula.
Step 5: Compute the approximate off-step values $y'_{n+\frac{1}{2}}$and $y_{n+\frac{1}{2}}$ using the off-step formula.
Step 6: Calculate the corrected values of $y'_{n+1}$and y_{n+1} applying the corrector formula.
Step 7: End the procedure.

NUMERICAL EXPERIMENTS AND COMPARISONS

To evaluate the efficiency of the proposed direct method, four second-order delay differential equations (DDE) problems with constant delay are considered as test cases. The numerical results obtained using the DAM1OS for Problems 1, 2, and 3 are compared with Haar Wavelet method, and Problem 4 is compared with the Laplace transform method.

The notations used in the tables are listed below:

N	Number of steps
DAM1OS	Direct Adams-Moulton one-off step method
Max Error	Maximum error (|approximate solutions-exact solutions)
TIME	Execution time in seconds

Problem 1.
Consider the second-order delay differential equation [Xuan et al. (2021)]:

$$y''(t) = -\frac{1}{2}y(t) + \frac{1}{2}y(t-\pi), t \in [0,\pi],$$

subject to the delay condition

$$y(t) = 1 - \sin(t), -\pi \leq t \leq 0,$$

and the initial conditions

$$y(0) = 1, y'(0) = -1.$$

The exact solution to this problem is

$$y(t) = 1 - \sin(t).$$

Problem 2.
Consider the second-order delay differential equation [Xuan et al. (2021)]:

$$y''(t) = y(t-\pi), t \in [0,\pi],$$

subject to the delay condition

$$y(t) = \sin(t), -\pi \leq t \leq 0,$$

and the initial conditions

$$y(0) = 0, y'(0) = 1.$$

The exact solution is

$$y(t) = \sin(t).$$

Problem 3.
Consider the second-order delay differential equation [Xuan et al. (2021)]:

$$y''(t) = -y(t) + y(t-1), t \in [0,2],$$

subject to the delay condition

$$y(t) = t^2 + 3t + 2, -1 \leq t \leq 0,$$

and the initial conditions

$$y(0) = 2, y'(0) = 0.$$

The exact solution is

$$y(t) = 4\cos(t) - \sin(t) + t^2 + t - 2.$$

Problem 4.
Consider the second-order delay differential equation [Aljawi et al, (2024)]:

$$y''(t) = -2y(t) - \frac{1}{2}y(t-1) + 2 + \frac{1}{2}(t-1)^2, t \in [0,1],$$

subject to the delay condition

$$y(t) = t^2, -1 \leq t \leq 0,$$

and the initial conditions

$$y(0) = 0, y'(0) = 0.$$

The exact solution is

$$y(t) = t^2.$$

Table 1: Comparison of the numerical results for Problem 1

		Haar Wavelet		DAM1OS	
j	$N = 2^{j+1}$	Max Error	Time	Max Error	Time
1	4	3.979(−03)	0.00158	5.263(−03)	0.021
2	8	1.048(−03)	0.00193	4.507(−04)	0.023
3	16	2.682(−04)	0.00461	3.554(−05)	0.026
4	32	6.777(−05)	0.01682	4.384(−06)	0.027
5	64	1.703(−05)	0.06525	8.856(−07)	0.030
6	128	4.269(−06)	0.20768	2.111(−07)	0.032
7	256	1.068(−06)	0.81604	5.218(−08)	0.031
8	512	2.673(−07)	3.27512	1.301(−08)	0.032
9	1024	7.685(−08)	13.0245	3.251(−09)	0.033

Table 2: Comparison of the numerical results for Problem 2

		Haar Wavelet		DAM1OS	
j	$N = 2^{j+1}$	Max Error	Time	Max Error	Time
1	4	4.269(−03)	0.00140	4.880(−03)	0.016
2	8	1.134(−03)	0.00166	5.212(−04)	0.022
3	16	2.917(−04)	0.00334	5.652(−05)	0.024
4	32	7.394(−05)	0.01211	9.012(−06)	0.027
5	64	1.861(−05)	0.03886	1.935(−06)	0.029
6	128	4.667(−06)	0.14378	4.657(−07)	0.032
7	256	1.168(−06)	0.53690	1.155(−07)	0.030

8	512	2.924(−07)	2.59874	2.884(−08)	0.031
9	1024	7.314(−08)	11.0023	7.213(−09)	0.034

Table 3: Comparison of the numerical results for Problem 3

		Haar Wavelet		DAM1OS	
j	$N = 2^{j+1}$	Max Error	Time	Max Error	Time
1	4	7.248(−03)	0.00197	3.810(−04)	0.015
2	8	1.952(−03)	0.00277	6.016(−05)	0.018
3	16	5.057(−04)	0.04996	9.302(−06)	0.021
4	32	1.286(−04)	0.01505	2.182(−06)	0.024
5	64	3.243(−05)	0.05394	5.217(−07)	0.026
6	128	8.142(−06)	0.19726	1.256(−07)	0.028
7	256	2.039(−06)	0.77262	3.066(−08)	0.031
8	512	5.105(−07)	3.07790	7.562(−09)	0.034
9	1024	1.276(−07)	12.3348	1.426(−09)	0.042

Table 4: Comparison of the numerical results for Problem 4

	Laplace transform		DAM1OS	
N	Max Error	Time	Max Error	Time
20	3.125(−10)	0.323	4.400(−16)	0.069
22	5.883(−10)	0.348	6.661(−16)	0.068
24	7.986(−11)	0.360	6.661(−16)	0.074
26	5.166(−12)	0.355	6.661(−16)	0.079
28	2.597(−14)	0.418	3.330(−16))	0.082
30	4.996(−14)	0.352	1.110(−16)	0.080
32	7.105(−15)	0.449	0.000(+00)	0.084
34	6.661(−16)	0.408	2.220(−16)	0.086
36	2.220(−16)	0.352	2.220(−16)	0.084

Tables 1–3 present the numerical solutions obtained by the proposed DAM1OS method for Problems 1–3, where the absolute errors at each point are compared with those produced by the

Haar wavelet method reported by Xuan et al. (2021). Table 4 provides the results for Problem 4, comparing the DAM1OS method with the Laplace transform method from Aljawi et al. (2024). The results demonstrate that the DAM1OS method consistently achieves smaller maximum absolute errors across various step sizes compared to both the Haar wavelet and Laplace transform methods. Furthermore, as shown in Tables 1–4, the absolute error decreases as the step size h is reduced (or equivalently, as N increases), indicating that the approximate solutions converge toward the exact solution with finer step sizes. In terms of computational efficiency, the proposed DAM1OS method also exhibits faster execution times than the comparison methods for certain test problems.

CONCLUSION

In this study, it has been demonstrated that the proposed DAM1OS method with a constant step size provides an effective and efficient approach for solving second-order delay differential equations (DDEs). A key advantage of the method lies in its ability to solve second-order DDEs in their original form, thereby eliminating the need to transform them into systems of first-order ordinary differential equations (ODEs). This direct formulation leads to a notable reduction in computational cost. Furthermore, the numerical results confirm that the proposed method achieves superior accuracy while maintaining lower computational complexity compared to existing techniques.

REFERENCES

Aljawi, S., Kamran, Hussain, S., Shah, K. and Abdeljawad, Th (2024). On the Numerical Solution of Second-order Delay Differential Equations via a Novel Approach. *J. Math. Computer Sci.*, **36**: 1–16.

Enright, W. H. and Hu, M. (1995). Interpolating Runge-Kutta Methods for Vanishing Delay Differential Equations. *Computing*, **55(3)**: 223–236.

Fatunla, S.O. (1991). Block Methods for Second-order ODEs. *International Journal of Computer Mathematics*, **41(1-2)**: 55–63.

Hoo, Y.S., Majid, Z.A., and Ismail, F. (2013). Solving Second-Order Delay Differential Equations by Direct Adams-Moulton Method. *Mathematical Problems in Engineering*, **2013(1)**: 261240.

Ishak, F. and Ahmad, S. N. (2011). Solving Delay Differential Equations by Predictor-Corrector Method using Lagrange and Hermite Interpolations. *2011 IEEE Colloquium on Humanities, Science and Engineering*, pp. 932–934. IEEE.

Ismail, N.I.N. and Majid, Z.A. (2024). Numerical solution on Neutral delay Volterra Integro-differential Equation. *Bulletin of the Malaysian Mathematical Sciences Society*, **47(3)**: 85.

Johari, M. A. M. and Majid, Z.A.(2022). Predictor-corrector Scheme for solving Second-order Ordinary Differential Equations. *Menemui Matematik (Discovering Mathematics)*, **44(2)**: 86–96.

Karoui, A. and Vaillancourt, R. (1995). A Numerical Method for Vanishing-lag Delay differential Equations. *Applied Numerical Mathematics*, **17(4)**: 383–395.

Lambert, J.D. (1973) Computational Methods in Ordinary Differential Equations. *John Wiley & Sons Inc.,* New-York.

Majid, Z.A. and Suleiman, M. (2006). Direct Integration Implicit Variable Steps Method for Solving Higher Order Systems of Ordinary Differential Equations Directly. *Sains Malaysiana*, **35(2)**: 63–68.

Majid, Z.A., Azmi, N. A. and Suleiman, M. (2009). Solving Second Order Ordinary Differential Equations *Using* Two Point Four Step Direct Implicit Block Method. *European Journal of Scientific Research*, **31(1)**: 29–36.

Majid, Z.A., See, P. P. and Suleiman, M. (2011). Solving Directly Two Point Non Linear Boundary Value Problems Using Direct Adams-Moulton Method. *Journal of Mathematics and Statistics*, **7(2)**: 124–128.

Rasdi, N. and Majid, Z.A.(2015). Solving Second-order Delay Differential Equations by Direct Extended Two and Three Point Implicit One-step Block Method. In *2015 International Conference on Research and Education in Mathematics (ICREM7)*, pp. 29–34. IEEE.

Sabir, Z., Guirao J. L., Saeed, T. and Erdogan, F. (2020). Design of a Novel Second-order Prediction Differential Model Solved by Using Adams and Explicit Runge–Kutta Numerical Methods. *Mathematical Problems in Engineering*, **2020(1)**: 9704968.

Suleiman, M. and Ismail, F. (2001). Solving Delay Differential Equations Using Componentwise Partitioning by Runge–Kutta Method. *Applied Mathematics and Computation*, **122(3)**: 301–323.

Xuan, Y., Amin, R., Zaman, F., Khan, Z., Ullah, I. and Nazir, S. (2021). Second-order Delay Differential Equations to Deal the Experimentation of Internet of Industrial Things via haar Wavelet Approach. *Wireless Communications and Mobile Computing*, **2021(1)**: 5551497.