Research Article

Implicit Two-Point Block Method for Solving Fourth-Order Initial Value Problem Directly with Application

Reem Allogmany,1,2 Fudziah Ismail,1,3 Zanariah Abdul Majid1,3 and Zarina Bibi Ibrahim1,3

1Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia
2Department of Mathematics, Faculty of Science, Taibah University, Al-Madinah Al-Munawarah, Saudi Arabia
3Institute for Mathematical Research, Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia

Correspondence should be addressed to Reem Allogmany; rlogmany@outlook.com

Received 18 December 2019; Revised 14 July 2020; Accepted 10 August 2020; Published 15 September 2020

Academic Editor: Stefan Balint

Copyright © 2020 Reem Allogmany et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes an implicit block method with two-point to directly solve the fourth-order Initial Value Problems (IVPs). The implicit block method is derived by adopting Hermite interpolating polynomial as the basis function, incorporating the first derivative of \( f(t, y, y', y'', y''') \) to enhance the solution’s accuracy. A block formulation is presented to acquire the numerical approximation at two points simultaneously. The introduced method’s basic properties, including order, zero stability, and convergence, are presented. Numerical experiments are carried out to verify the accuracy and efficiency of the proposed method compared with those of the several existing methods. Application in ship dynamics is also presented which yield impressive results for the proposed two-point block method.

1. Introduction

Higher order differential equations (ODEs) can be used to model problems arising from the field of applied sciences and engineering [1]. The problem of static deflection of a uniform beam which can be modelled as a fourth-order initial value problem (IVP) is a good example of a real problem in engineering [2, 3]. The classical way of solving them is by reducing the equation into the system of first-order ODEs, but this process is too rigorous compared with the direct methods [4, 5]. In addition, many researchers have presented direct methods to avoid the reduction effort [6–15]. To enhance the efficacy of numerical methods, the block method is introduced with the idea of producing simultaneously \( r \)-point of the approximate solutions at one time step. Kuboye and Omar [7] and Jacob [8] introduced efficient zero-stable numerical block methodologies for fourth-order initial value problems (IVPs). For block multistep method, approximations are used to generate the approximate solutions at the new block. Omar and Kuboye [9] proposed a block method for directly solving general fourth-order IVPs by increasing the step number; Adeyeye and Omar [14] also presented a direct six-step block method. Recently, researchers have been exploring numerical methods with more functional evaluations with the aim of obtaining numerical solutions with very high precision. This is how hybrid methods in nature arises. Jator [16] and Yap and Ismail [10] developed four-step hybrid block methods, and Abdelrahim and Omar [11] generalized three off-step points. Kayode et al. [12] modified the implicit hybrid block method to directly solve the IVPs associated with fourth-order ODEs. Generally, collocation and interpolation techniques are utilized as direct methods. The points need to be collocated and interpolated after which a system of linear equations must be resolved in order to obtain the method’s coefficients. Therefore, we developed a two-point block the implicit method by using a strategy in our proposed method which can be implemented in a straightforward manner. The method replaces the function by interpolating and integrating the polynomial. Our derivation only requires us to
do interpolation and integration to obtain the coefficients. The method also involves the fifth derivative of the solution to equation (1), which aims at acquiring better accuracy. In general, the accuracy of a method increases with increase in the order of the method. But, with the idea of incorporation of higher derivative of the solution in the process, higher and better accuracy is achieved without corresponding increase in the order of the method. We are concerned with the development of the numerical solution of IVPs for fourth-order ODEs of the form:

\[ \begin{align*}
  y^{(4)} &= f(t, y, y', y'', y''') \\
y(a) &= y_0, \\
y'(a) &= y'_0, \\
y''(a) &= y''_0, \\
y'''(a) &= y'''_0,
\end{align*} \tag{1} \]

where the fifth derivative of the solution to equation (1) is

\[ y^{(5)} = f'(t, y, y', y'', y''') + f f y'' + f y''' + g(t, y, y', y'', y'''). \tag{2} \]

We assume that \( f \) in equation (1) is differentiable to a desired order in region \( R \) and \( f(t, y, y', y'', y''') \) satisfies Lipchitz condition in its second, third, fourth, and fifth terms as follows:

\[ \begin{align*}
  |f(t, y_1, y', y'', y''') - f(t, y_2, y', y'', y''')| &\leq L_1 |y_1 - y_2|, \\
  |f(t, y_1, y', y'', y''') - f(t, y, y', y'', y''')| &\leq L_2 |y_1 - y_2|, \\
  |f(t, y, y', y'', y''') - f(t, y, y', y'', y''')| &\leq L_3 |y_1 - y_2|, \\
  |f(t, y, y', y'', y''')| &\leq L_4 |y_1 - y_2|,
\end{align*} \tag{3} \]

for all points \((t, y_1, y', y'', y'''), (t, y_2, y', y'', y'''), (t, y, y', y'', y'''), (t, y, y', y'', y'''), i = 1, 2, \) in the region \( R \). Then, the IVPs in equation (1) have a unique solution in \( R \) ([17, 18]).

The paper is structured in the following manner: in Section 2, a discussion is undertaken about the derivation of the block method of order six (I2PBD06) along with the basic idea on how the block works. The implementation of this method to solve general fourth-order ODEs is proposed in Section 3. Numerical experiments and the application of the ship dynamics problem are given in Section 4 to show the accuracy and efficiency of the proposed method. The study’s conclusion is finally provided in Section 5.

2. Derivation of Method

This section depicts the proposed method’s derivation based on Hermite Interpolating Polynomial \( P_5(t) \), which interpolates \( f(t, y, y', y'', y''') \) at two points. The polynomial has the form

\[ P(t) = \sum_{i=0}^{n} \prod_{k=0}^{m-1} f^{(k)} L_{i,k}(t), \tag{4} \]

where

\[ f_i = f\left(t_i, y_i, y'_i, y''_i, y'''_i\right), \]

\[ t_i = a + i h, \]

\[ i = 0, 1, \ldots, n, \]

and \( h = (b - a)/n \); \( L_{i,k}(t) \) is the generalized Lagrange polynomial, \( i = 0, 1, \ldots, n, k = 0, 1, \ldots, m_i \), and \( n \) is a positive integer. The method computes the approximate solutions \( y_{n+1} \) and \( y_{n+2} \) concurrently at \( t_{n+1} \) and \( t_{n+2} \), where \( t_n \) becomes the starting point and \( t_{n+2} \) is the last point in the block with step size \( 2h \). The evaluation solution of \( y_{n+2} \) at the point \( t_{n+2} \) will be used as the next iteration’s initial value:

\[ \begin{align*}
P_2(t) &= L_{0,0}(t)f_0 + L_{1,0}(t)f_1 + L_{2,0}(t)f_2 + L_{0,1}(t)g_0 \\
&\quad + L_{1,1}(t)g_1 + L_{2,1}(t)g_2, \tag{6} \end{align*} \]

where \( g \) is the fifth derivative of the solution and

\[ \begin{align*}
L_{0,0}(t) &= \left( \frac{t - t_{n+1}}{t_n - t_{n+1}} \right)^2 \left( \frac{t - t_{n+2}}{t_n - t_{n+2}} \right)^2 \left( \frac{2}{t_n - t_{n+2}} + \frac{2}{t_n - t_{n+1}} \right), \\
L_{1,0}(t) &= \left( \frac{t - t_{n+1}}{t_n - t_{n+1}} \right)^2 \left( \frac{t - t_{n+2}}{t_n - t_{n+2}} \right)^2 \left( \frac{2}{t_n - t_{n+2}} + \frac{2}{t_n - t_{n+1}} \right), \\
L_{2,0}(t) &= \left( \frac{t - t_{n+1}}{t_n - t_{n+1}} \right)^2 \left( \frac{t - t_{n+2}}{t_n - t_{n+2}} \right)^2 \left( \frac{2}{t_n - t_{n+2}} + \frac{2}{t_n - t_{n+1}} \right), \\
L_{0,1}(t) &= \left( \frac{t - t_{n+1}}{t_n - t_{n+1}} \right)^2 \left( \frac{t - t_{n+2}}{t_n - t_{n+2}} \right)^2 \left( \frac{2}{t_n - t_{n+2}} + \frac{2}{t_n - t_{n+1}} \right), \\
L_{1,1}(t) &= \left( \frac{t - t_{n+1}}{t_n - t_{n+1}} \right)^2 \left( \frac{t - t_{n+2}}{t_n - t_{n+2}} \right)^2 \left( \frac{2}{t_n - t_{n+2}} + \frac{2}{t_n - t_{n+1}} \right), \\
L_{2,1}(t) &= \left( \frac{t - t_{n+1}}{t_n - t_{n+1}} \right)^2 \left( \frac{t - t_{n+2}}{t_n - t_{n+2}} \right)^2 \left( \frac{2}{t_n - t_{n+2}} + \frac{2}{t_n - t_{n+1}} \right). \tag{7} \end{align*} \]
The approximate solution $y_{n+1}$ at the point $t_{n+1}$ can be retrieved by integrating equation (1) once, twice, thrice, and four times, in relation to $t$ over the interval $[t_n, t_{n+1}]$. The following formula can be reached:

\[
\int_{t_n}^{t_{n+1}} y^{(4)} \, dt = \int_{t_n}^{t_{n+1}} f(t, y, y', y'', y''') \, dt,
\]

\[
\int_{t_n}^{t_{n+1}} \int_{t_n}^{t} y^{(4)} \, dt \, dt = \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} f(t, y, y', y'', y''') \, dt \, dt,
\]

\[
\int_{t_n}^{t_{n+1}} \int_{t_n}^{t} \int_{t_n}^{t} y^{(4)} \, dt \, dt \, dt = \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} \int_{t_n}^{t} f(t, y, y', y'', y''') \, dt \, dt \, dt.
\]

Let $t_{n+1} = t_n + h$ which gives

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

\[
y''(t_{n+1}) = y''(t_n) + hy'''(t_n) + \int_{t_n}^{t_{n+1}} [f(t_n - t) f(t, y, y', y'', y''')] \, dt,
\]

\[
y'(t_{n+1}) = y'(t_n) + hy''(t_n) + \frac{h^2}{2} y'''(t_n) + \int_{t_n}^{t_{n+1}} \left( \frac{t_{n+1} - t}{2} \right) \left( f(t, y, y', y'', y''') \right) \, dt,
\]

\[
y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2} y''(t_n) + \frac{h^3}{6} y'''(t_n) + \int_{t_n}^{t_{n+1}} \left( \frac{t_{n+1} - t}{2} \right)^2 \left( f(t, y, y', y'', y''') \right) \, dt.
\]

Then, $f(t, y, y', y'', y''')$ in equations (9)–(12) will be replaced by Hermite interpolating polynomial $P_2(t)$. Taking $t = t_{n+1} + sh$ leads to $s = (t - t_{n+2}/h)$ and $dt = hds$ and then making changes in the integration limit from $-2$ to $-1$, we have

\[
y'''(t_{n+1}) = y'''(t_n) + \int_{-2}^{-1} \left[ f_0 L_{0,0}(s) + f_1 L_{1,0}(s) + f_2 L_{2,0}(s) + g_0 L_{0,1}(s) + g_1 L_{1,1}(s) + g_2 L_{2,1}(s) \right] hds,
\]

\[
y''(t_{n+1}) = y''(t_n) + hy'''(t_n) + \int_{-2}^{-1} (-h(s + 1)) \left[ f_0 L_{0,0}(s) + f_1 L_{1,0}(s) + f_2 L_{2,0}(s) + g_0 L_{0,1}(s) + g_1 L_{1,1}(s) + g_2 L_{2,1}(s) \right] hds,
\]

\[
y'(t_{n+1}) = y'(t_n) + hy''(t_n) + \frac{h^2}{2} y'''(t_n) + \int_{-2}^{-1} \frac{(h(s + 1))^2}{2} \left[ f_0 L_{0,0}(s) + f_1 L_{1,0}(s) + f_2 L_{2,0}(s) + g_0 L_{0,1}(s) + g_1 L_{1,1}(s) + g_2 L_{2,1}(s) \right] hds,
\]

\[
y(t_{n+1}) = y(t_n) + hy'(t_n) + \frac{h^2}{2} y''(t_n) + \frac{h^3}{6} y'''(t_n) + \int_{-2}^{-1} \frac{(-h(s + 1))^3}{6} \left[ f_0 L_{0,0}(s) + f_1 L_{1,0}(s) + f_2 L_{2,0}(s) + g_0 L_{0,1}(s) + g_1 L_{1,1}(s) + g_2 L_{2,1}(s) \right] hds,
\]
where

\[ L_{0,0} = \frac{1}{4} (s + 1)^3 s^2 (1 + 3 (2 + s)), \]
\[ L_{1,0} = (2 + s)^3 s^2, \]
\[ L_{2,0} = \frac{1}{4} (2 + s)^3 (s + 1)^2 (1 - 3s), \]

(17)
\[ L_{0,1} = \frac{h}{4} (2 + s) (s + 1)^3 s^2, \]
\[ L_{1,1} = h (s + 1) (2 + s)^3 s^2, \]
\[ L_{2,1} = \frac{h}{4} s (s + 1)^3 (2 + s)^2. \]

Evaluating the integrals in equations (13)–(16) gives

\[ y_{n+1}^m = y_n^m + h \left[ \frac{101}{240} f_n + \frac{128}{240} f_{n+1} + \frac{11}{240} f_{n+2} \right] + h^2 \left[ \frac{13}{240} g_n - \frac{40}{240} g_{n+1} - \frac{3}{240} g_{n+2} \right], \]  
(18)
\[ y_{n+1}^m = y_n^m + h y_n'' + h^2 \left[ \frac{13}{42} f_n + \frac{7}{42} f_{n+1} + \frac{1}{42} f_{n+2} \right] + h^3 \left[ \frac{59}{1680} g_n - \frac{128}{1680} g_{n+1} - \frac{11}{1680} g_{n+2} \right], \]  
(19)
\[ y_{n+1}^m = y_n^m + h y_n'' + h^2 \left[ \frac{817}{6720} f_n + \frac{256}{6720} f_{n+1} + \frac{47}{6720} f_{n+2} \right] + h^3 \left[ \frac{83}{6720} g_n - \frac{140}{6720} g_{n+1} - \frac{13}{6720} g_{n+2} \right], \]  
(20)
\[ y_{n+1}^m = y_n^m + h y_n'' + h^2 \left[ \frac{67}{2016} f_n + \frac{14}{2016} f_{n+1} + \frac{3}{2016} f_{n+2} \right] + h^3 \left[ \frac{185}{60480} g_n - \frac{256}{60480} g_{n+1} - \frac{25}{60480} g_{n+2} \right]. \]  
(21)

Now, we have four formulas, one for the approximate solution and the others for the approximation of the first, second, and third derivatives of the proposed solution at the block’s end point \([t_n, t_{n+1}].\) Therefore, we need to consider the evaluation of \(p_2(t)\) at the point \(y_{n+2}\) over \([t_n, t_{n+2}].\) By applying the same technique as for the formula for \(y_{n+1},\) we have the following formula at \(t_{n+2}:

\[ y_{n+2}^m = y_{n+1}^m + h \left[ \frac{11}{240} f_n + \frac{128}{240} f_{n+1} + \frac{101}{240} f_{n+2} \right] + h^2 \left[ \frac{3}{240} g_n + \frac{40}{240} g_{n+1} - \frac{13}{240} g_{n+2} \right], \]  
(22)
\[ y_{n+2}^m = y_{n+1}^m + h y_{n+1}'' + h^2 \left[ \frac{37}{1680} f_n + \frac{616}{1680} f_{n+1} + \frac{187}{1680} f_{n+2} \right] + h^3 \left[ \frac{5}{840} g_n + \frac{76}{840} g_{n+1} + \frac{16}{840} g_{n+2} \right], \]  
(23)
\[ y_{n+2}^m = y_{n+1}^m + h y_{n+1}'' + h^2 \left[ \frac{41}{6720} f_n + \frac{928}{6720} f_{n+1} + \frac{151}{6720} f_{n+2} \right] + h^3 \left[ \frac{11}{6720} g_n + \frac{188}{6720} g_{n+1} - \frac{29}{6720} g_{n+2} \right], \]  
(24)
\[ y_{n+2}^m = y_{n+1}^m + h y_{n+1}'' + h^2 \left[ \frac{5}{4032} f_n + \frac{148}{4032} f_{n+1} + \frac{15}{4032} f_{n+2} \right] + h^3 \left[ \frac{10}{30240} g_n + \frac{190}{30240} g_{n+1} - \frac{23}{30240} g_{n+2} \right]. \]  
(25)

2.1. Order of the Method. The equations (15)–(22) can be written in a matrix difference equation to obtain the order of the new method as follows:

\[ \begin{align*}
\alpha Y_m &= h \beta Y_m' + h^2 \gamma Y_m'' + h^3 \delta Y_m''' + h^4 \zeta Y_m''', \\
\end{align*} \]  
(26)
where \(\beta, \gamma,\) and \(\delta\) are defined as
\[
\begin{align*}
\alpha &= \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 1
\end{bmatrix}, &
\beta &= \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}, \\
\gamma &= \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{bmatrix}, &
\delta &= \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}, \\
\zeta &= \begin{bmatrix}
101 & 128 & 11 & 0 \\
13 & 7 & 1 & 0 \\
817 & 256 & 47 & 0 \\
67 & 14 & 3 & 0 \\
11 & 128 & 101 & 0 \\
37 & 11 & 187 & 0 \\
41 & 928 & 151 & 0 \\
5 & 148 & 15 & 0
\end{bmatrix}, &
\lambda &= \begin{bmatrix}
13 & -40 & -3 & 0 \\
59 & -128 & -11 & 0 \\
83 & -140 & -13 & 0 \\
185 & -256 & -25 & 0 \\
3 & 40 & -13 & 0 \\
5 & 76 & -16 & 0 \\
11 & 188 & -29 & 0 \\
10 & 190 & -23 & 0
\end{bmatrix}
\end{align*}
\]
It is possible to define the linear operator that is linked to equation (26) as
\[
\ell[y(t)] = h \sum_{i=0}^{\infty} c_i y^{(i)}(t).
\]

Expanding equation (28) in the Taylor series yields
\[
\ell[y(t)] = C_0 y(t) + C_1 y'(t) + C_2 y''(t) + C_3 y'''(t) + \cdots + C_p y^{(p)}(t) + C_{p+1} y^{(p+1)}(t) + \cdots.
\]

The linear operator and the new method have order \( p \) if \( C_0 = C_1 = \cdots = C_p = C_{p+1} = 0 \), which is the error constant. Hence, in our method, \( C_0 = C_1 = \cdots = C_9 = 0 \) and \( C_{10} = [1.058199 \times 10^{-4}, 5.78707 \times 10^{-5}, 1.763674 \times 10^{-5}, 3.858 \times 10^{-6}, 1.0581959 \times 10^{-4}, 4.7950011 \times 10^{-5}, 1.267633 \times 10^{-5}, 2.4798443 \times 10^{-6}]^T \). Thus, it can be concluded that the order of the proposed method is 6.

2.2. Zero Stability. The method (18)–(25) is deemed zero-stable in case the roots \( z_i = 1, 2, \ldots, N \) of the first characteristic polynomial \( \rho(z) = \det(zA^{(0)} - A^{(1)}) \) is found to satisfy \( |z_i| \leq 1 \).

Now, in our proposed method, we will use the following technique to find the matrix of the first characteristic polynomial.

By replacing equation (18) into equation (22), we have
\[
y_n'' = y_n'' + h^2 \left[ 1264 \frac{f_n}{1680} + 1792 \frac{f_n}{1680} + 304 \frac{f_{n+1}}{1680} \right] + h^3 \left[ 16 \frac{g_n}{240} - 16 \frac{g_{n+1}}{240} \right].
\]

Also, substituting equations (18) and (19) into equation (23), we get
\[
y_n'' = y_n'' + 2h^2 y_n'' + h^3 \left[ 1264 \frac{f_n}{1680} + 1792 \frac{f_n}{1680} + 304 \frac{f_{n+1}}{1680} \right] + h^4 \left[ 105g_n - 16 \frac{g_{n+1}}{105} - 4 \frac{g_{n+2}}{105} \right].
\]

Substituting equations (18)–(21) into equation (24), we have
\[
y_n' = y_n' + 2h y_n'' + h^2 \left[ 68 \frac{f_n}{105} + 64 \frac{f_n}{105} + 8 \frac{f_{n+1}}{105} \right] + h^3 \left[ 512 \frac{g_n}{6720} - 1165 \frac{g_n}{6720} - 128 \frac{g_{n+1}}{6720} \right].
\]
\[ y_{n+2} = y_n + 2\frac{h}{63} y'_n + 2\frac{h}{63} y''_n + 4\frac{h}{24} y'''_n + h^4 \]

\[ \cdot \left[ \frac{24}{63} f_n + \frac{16}{63} f_{n+1} + \frac{2}{63} f_{n+2} \right] + h^5 \]  

(33)

By using the equations (18)–(22) and (30)–(33), the general form of the matrix \( A^{(0)} \) and \( A^{(1)} \) can be written as \( A^{(0)} = 8 \times 8 \), identity matrix,

\[ A^{(1)} = \begin{bmatrix} 
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 
\end{bmatrix}. \]

(34)

Then, \( \rho(z) = det(z A^{(0)} - A^{(1)}) = 0 \) implies that \( \rho(z) = det z(z - 1)^4 \). Hence, \( z = 0, 0, 0, 0, 1, 1, 1 \)

Therefore, it can be inferred that the proposed method has zero stability.

3. Implementation

In this section, we briefly mention the developed method’s implementation. The approximate solutions in equations (15)–(22) will be estimated using the predictor-corrector schemes. The predictor equations using Taylor method are

\[ y^{p,c}_{n+1} = y^{c}_{n+1} + hf^{p}_{n+1}, \]

\[ y^{p,c}_{n+1} = y^{c}_{n+1} + hy^{p,c}_{n+1} + \frac{h^2}{2!} f^{p}_{n+1}, \]

\[ y^{p,c}_{n+1} = y^{c}_{n+1} + hy^{p,c}_{n+1} + \frac{h^3}{3!} f^{p}_{n+1}, \]

\[ y^{p,c}_{n+1} = y^{c}_{n+1} + hy^{p,c}_{n+1} + \frac{h^4}{4!} f^{p}_{n+1} \]

(35)

Define equation (35) as the initial approximation and apply the derived method in equations (18)–(25) as the corrector. In order to obtain the corrector iteration, the following equations will be used

\[ y^{c}_{n+1} = y^{c}_{n} + h y^{c}_{n} + \frac{h^2}{2} y^{c}_{n} + \frac{h^3}{3!} y^{c}_{n} + \frac{h^4}{4!} y^{c}_{n} + \frac{h^5}{5!} y^{c}_{n} \]

(36)
Then, we evaluate the functions $f_{y_n}^c$ and $g_{y_n}^c$ which we use to compute the following approximate solutions at the second point:

\[
y_{n+2}^c = y_{n+1}^c + \frac{h}{240} [11 f_{y_n}^c + 128 f_{y_{n+1}}^c + 101 f_{y_{n+2}}^c] + \frac{h^2}{240} [3 g_{y_n}^c + 40 g_{y_{n+1}}^c - 13 g_{y_{n+2}}^c],
\]

\[
y_{n+2}^c = y_{n+1}^c + h y_{n+1}^c + \frac{h^2}{1680} [37 f_{y_n}^c + 616 f_{y_{n+1}}^c + 187 f_{y_{n+2}}^c] + \frac{h^3}{840} [5 g_{y_n}^c + 76 g_{y_{n+1}}^c + 16 g_{y_{n+2}}^c],
\]

\[
y_{n+2}^c = y_{n+1}^c + h y_{n+1}^c + \frac{h^2}{6720} [41 f_{y_n}^c + 928 f_{y_{n+1}}^c + 151 f_{y_{n+2}}^c] + \frac{h^3}{6720} [11 g_{y_n}^c + 188 g_{y_{n+1}}^c - 29 g_{y_{n+2}}^c],
\]

\[
y_{n+2}^c = y_{n+1}^c + h y_{n+1}^c + \frac{h^2}{31! y_{n+1}}^c + \frac{h^3}{4032} [5 f_{y_n}^c + 148 f_{y_{n+1}}^c + 15 f_{y_{n+2}}^c] + \frac{h^4}{30240} [10 g_{y_n}^c + 190 g_{y_{n+1}}^c - 23 g_{y_{n+2}}^c].
\]

Next, we evaluate the functions $f_{y_n}^c$ and $g_{y_n}^c$ that we will use in the next corrector iteration. Hence, we repeat the procedure given in equations (36) and (37) to perform the next corrector iterations until the end of the interval.

4. Results and Discussion

Within this section, we tested the I2PBDO6 method to solve single as well as a system of initial value challenges in the form of equation (1). We will see how the method of two-point block (order 6) produces comparable results to the existing direct methods of similar characteristics. Most of the existing methods are of higher order in comparison to the proposed method. The following notations will be used in the tables:

(i) I2PBDO6: the direct implicit two-point block method proposed in this paper, order 6
(ii) MCM: direct maximal order multiderivative collocation method [6]
(iii) ZSB7: the direct zero-stable block method of order seven [7]
(iv) ZSDM6: the direct zero-stable method of order six [8]
(v) IHB8: the direct implicit hybrid block method of order eight [12]
(vi) FDM6: the direct linear multistep method of order six [16]
(vii) BHC8: the direct block hybrid collocation method of order eight [10]
(viii) FSHB8: the direct four-step hybrid block method of order eight [11]
(ix) SSB8: the direct seven-step block method of order eight [9]
(x) LIBO7: the direct implicit six-step block method of order seven [14]
(xi) RKF5: the direct RKTF method of order five with four-stage [13]

\[
\begin{align*}
\text{(xii)} & \quad \text{RKF5: the RK method with six-stage given in Lambert [19]} \\
\text{(xiii)} & \quad h: \text{step size} \\
\text{(xiv)} & \quad \text{FCN: number of function evaluations} \\
\text{(xv)} & \quad \text{NS: number of steps} \\
\text{(xvi)} & \quad \text{MAXE: maximum absolute error}
\end{align*}
\]

4.1. Problems

Problem 1

\[
y^{(4)} + y'' = 0, \quad 0 \leq t \leq \frac{\pi}{2},
\]

\[
y(0) = 0,
\]

\[
y'(0) = \frac{-1.1}{72 - 50\pi},
\]

\[
y''(0) = \frac{1}{144 - 100\pi},
\]

\[
y'''(0) = \frac{1.2}{144 - 100\pi}.
\]

Exact solution:

\[
y(t) = \frac{1 - t - \cos(t) - 1.2\sin(t)}{144 - 100\pi}.
\]

Problem 2

\[
y^{(4)} = y'^2 - 4y''' - 4t^2 + e^t(1 - 4t + t^2), \quad 0 \leq t \leq 1,
\]

\[
y(0) = 1,
\]

\[
y'(0) = 1,
\]

\[
y''(0) = 3,
\]

\[
y'''(0) = 1.
\]
Exact solution:

\[ y(t) = t^2 + e^t. \quad (41) \]

**Problem 3**

\[ y^{(4)} = y^{'''} + y'' + 2y, \quad t \in [0, 2], \]

\[ y(0) = 0, \]

\[ y'(0) = 0, \]  \hspace{1cm} (42)

\[ y''(0) = 0, \]

\[ y'''(0) = 30. \]

Exact solution:

\[ y(t) = 2e^{2t} - 5e^{-t} + 3\cos(t) - 9\sin(t). \quad (43) \]

**Problem 4**

\[ y^{(4)} - \left(t^4 + 14t^3 + 49t^2 + 32t - 12\right)e^t = 0, \]

\[ y(0) = 0, \]

\[ y'(0) = 0, \]  \hspace{1cm} (44)

\[ y''(0) = 2, \]

\[ y'''(0) = -6. \]

Exact solution:

\[ y(t) = t^2 (1 - t)^3 e^t. \quad (45) \]

**Problem 5**

\[ y^{(4)} = -ty - \left(8 + 7t + t^3\right)e^t, \quad t \in [0, 2], \]

\[ y(0) = 0, \]

\[ y'(0) = 1, \]  \hspace{1cm} (46)

\[ y''(0) = 0, \]

\[ y'''(0) = -3. \]

Exact solution:

\[ y(t) = t (1 - t)e^t. \quad (47) \]

\[ y_1(0) = 1, \]

\[ y''_1(0) = 1, \]

\[ y'''_1(0) = 1, \]

\[ y^{(4)}_1 = -y''_1, \]

\[ y_2(0) = 1, \]

\[ y''_2(0) = -1, \]

\[ y'''_2(0) = -1, \]

\[ y^{(4)}_2 = -y''_1, \]

\[ y_3(0) = 1, \]

\[ y''_3(0) = -1, \]

\[ y'''_3(0) = -1, \]

\[ y^{(4)}_3 = -y''_1 - y_3 - \cos(t), \]

\[ y_4(0) = 1, \]

\[ y''_4(0) = -1, \]

\[ y'''_4(0) = -1, \]

\[ y^{(4)}_4 = -y''_1 - y_4 - 2\cos(t). \]

**Problem 6**

\[ y_1^{(4)} = -y_2'', \]

\[ y_2(0) = 0, \]

\[ y''_2(0) = 0, \]

\[ y'''_2(0) = 0, \]

\[ y^{(4)}_2 = -y''_1, \]

\[ y_3(0) = 0, \]

\[ y''_3(0) = 0, \]

\[ y'''_3(0) = 0, \]

\[ y^{(4)}_3 = -y''_1 - y_3 - \cos(t), \]

\[ y_4(0) = 0, \]

\[ y''_4(0) = 0, \]

\[ y'''_4(0) = 0, \]

\[ y^{(4)}_4 = -y''_1 - y_4 - 2\cos(t). \]

**Problem 7**

Exact solution:

\[ y_1(t) = e^t, \]

\[ y_2(t) = -e^t, \]

\[ y_3(t) = -\cos(t), \]

\[ y_4(t) = -2\cos(t). \]  \hspace{1cm} (49)
\[ y_1^{(4)} = e^{3t} y_4, \]
\[ y_1(0) = 1, \]
\[ y_1'(0) = -1, \]
\[ y_1''(0) = 1, \]
\[ y_1'''(0) = -1, \]
\[ y_2^{(4)} = 16 e^{-t} y_1, \]
\[ y_2(0) = 1, \]
\[ y_2''(0) = -2, \]
\[ y_2'''(0) = 4, \]
\[ y_2''''(0) = -8, \]
\[ y_3^{(4)} = 81 e^{-t} y_2, \]
\[ y_3(0) = 1, \]
\[ y_3'(0) = -3, \]
\[ y_3''(0) = 9, \]
\[ y_3'''(0) = -27, \]
\[ y_4^{(4)} = 256 e^{-t} y_3, \]
\[ y_4(0) = 1, \]
\[ y_4'(0) = -4, \]
\[ y_4''(0) = 16, \]
\[ y_4'''(0) = -64. \] (50)

Exact solution:
\[ y_1(t) = e^{-t}, \]
\[ y_2(t) = e^{-2t}, \]
\[ y_3(t) = e^{-3t}, \]
\[ y_4(t) = e^{-4t}, \] (51)
\[ 0 \leq t \leq 3. \]

The aim of this study was to demonstrate the accuracy, efficiency, and applicability of the proposed I2PBDO6 method. Tables 1–4 show the absolute errors at different points as well as different step size. It is remarkable that the errors are zero at some points of the interval which means that the computed solutions are identical to the exact solutions with 20 digits at least. The two-point block method can be seen to significantly outperform MCM [6], ZSB7 [7], LIBO7 [14], ZSDM6 [8], and IHB8 [12] with regard to accuracy. For Problems 3 to 5, the results of the proposed I2PBDO6 method are tabulated alongside others in Tables 5–7. We have considered the absolute errors along the interval using different number of steps in terms of accuracy as well as steps taken. It can be observed that increasing the number of steps leads to decrease in the error except that in Table 6 at the number of step 5004 and 50004, since Problem 4 involves exponential function which requires an infinite number of digits for numerical representation. However, these numbers must be represented by a finite number of digits in computer for numerical computation that causes round-off errors. Thus, at a large number of steps such as 5004 and 50004, a sequence of many operations and each subject to a small rounding error is obtained, where the error accumulates and affects the accuracy of the final solutions. The proposed method shows very good performance compared with FDM6 [16], BHC8 [10], FSHB8 [11], SSB8 [9], and LIBO7 [14]. The system in Problem 6 and Problem 7 has integrated in the interval \( t \in [0, 10] \) and \( t \in [0, 3] \), respectively. A comparison of numerical outcomes is made with the solution of the same group of systems when they are solved by using RKTFS5 [13] and RKF5 [19] methods. We have solved these systems to consider the maximum errors and the number of function evaluations at different step sizes.

Table 1: Numerical findings to solve Problem 1 with step size \( h = (1/320) \).

<table>
<thead>
<tr>
<th>( t )</th>
<th>MCM [6]</th>
<th>I2PBDO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.013125</td>
<td>0.21580746667 − 12</td>
<td>2.165587 − 19</td>
</tr>
<tr>
<td>0.20625</td>
<td>0.23563428636 − 12</td>
<td>2.162928 − 18</td>
</tr>
<tr>
<td>0.30625</td>
<td>0.250504044591 − 12</td>
<td>6.913642 − 18</td>
</tr>
<tr>
<td>0.40625</td>
<td>0.258852835000 − 12</td>
<td>1.122300 − 17</td>
</tr>
<tr>
<td>0.50625</td>
<td>0.25795515601 − 12</td>
<td>1.466220 − 17</td>
</tr>
<tr>
<td>0.603125</td>
<td>0.245233520990 − 12</td>
<td>4.308768 − 18</td>
</tr>
<tr>
<td>0.703125</td>
<td>0.216478744652 − 12</td>
<td>8.610773 − 18</td>
</tr>
<tr>
<td>0.803125</td>
<td>0.168292463298 − 12</td>
<td>1.893054 − 17</td>
</tr>
<tr>
<td>0.903125</td>
<td>0.969693075836 − 13</td>
<td>2.923863 − 17</td>
</tr>
</tbody>
</table>

Table 2: Numerical findings to solve Problem 1 with step size \( h = 0.01 \).

<table>
<thead>
<tr>
<th>( t )</th>
<th>ZSB7 [7]</th>
<th>I2PBDO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>5.421011 − 20</td>
<td>5.420312 − 20</td>
</tr>
<tr>
<td>0.02</td>
<td>5.421011 − 20</td>
<td>5.419616 − 20</td>
</tr>
<tr>
<td>0.03</td>
<td>2.710505 − 19</td>
<td>2.709462 − 19</td>
</tr>
<tr>
<td>0.04</td>
<td>1.084202 − 19</td>
<td>0.000000 + 00</td>
</tr>
<tr>
<td>0.05</td>
<td>3.252607 − 19</td>
<td>3.250530 − 19</td>
</tr>
<tr>
<td>0.06</td>
<td>3.252607 − 19</td>
<td>3.250120 − 19</td>
</tr>
<tr>
<td>0.08</td>
<td>1.734723 − 18</td>
<td>0.000000 + 00</td>
</tr>
<tr>
<td>0.09</td>
<td>4.336809 − 18</td>
<td>2.165937 − 19</td>
</tr>
<tr>
<td>0.10</td>
<td>8.456777 − 18</td>
<td>0.000000 + 00</td>
</tr>
</tbody>
</table>

Table 3: Numerical findings to solve Problem 1 with step size \( h = 0.1 \).

<table>
<thead>
<tr>
<th>( t )</th>
<th>LIBO7 [14]</th>
<th>I2PBDO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>6.51 − 19</td>
<td>0.000000 + 00</td>
</tr>
<tr>
<td>0.2</td>
<td>8.67 − 19</td>
<td>8.652339 − 19</td>
</tr>
<tr>
<td>0.3</td>
<td>4.34 − 19</td>
<td>4.321318 − 19</td>
</tr>
<tr>
<td>0.4</td>
<td>1.73 − 18</td>
<td>1.590085 − 18</td>
</tr>
<tr>
<td>0.5</td>
<td>8.67 − 19</td>
<td>3.450126 − 19</td>
</tr>
<tr>
<td>0.6</td>
<td>0.00 + 00</td>
<td>1.723552 − 19</td>
</tr>
<tr>
<td>0.7</td>
<td>0.00 + 00</td>
<td>0.000000 + 00</td>
</tr>
<tr>
<td>0.8</td>
<td>0.00 + 00</td>
<td>0.000000 + 00</td>
</tr>
<tr>
<td>0.9</td>
<td>0.00 + 00</td>
<td>0.000000 + 00</td>
</tr>
<tr>
<td>1.0</td>
<td>0.00 + 00</td>
<td>1.719064 − 19</td>
</tr>
</tbody>
</table>
The proposed I2PBDO6 method achieved smaller error for the same number of function evaluations compared with the I2PBDO6 method is improving clearly when smaller sizes, as shown in Tables 8 and 9. As we can see, the efficiency of the I2PBDO6 method is improving clearly when smaller step sizes are used. Based on the efficiency curves in Figures 1 and 2 of log (max error) versus log (function evaluations), the proposed I2PBDO6 method achieved smaller error for the same number of function evaluations compared with RKT5F and RK5F. Hence, I2PBDO6 accumulated more total function calls; nevertheless, Figures 1 and 2 display the best performances and efficiency of the proposed I2PBDO6 method.

Overall, the proposed method I2PBDO6 of order 6 has shown remarkable convergence since the approximate solution is very close to the exact solution. Moreover, the method is more efficient compared with the all present methods whose order is nearly equal to or greater as compared with that of the proposed method. The data given in Tables 1–9 and Figures 1 and 2 show the superiority of I2PBDO6 with regard to efficiency, accuracy, and total number of steps taken at different points of $t$ or different step sizes.

### 4.2. Application to Ship Dynamics Problem

We take into consideration a physical challenge from the ship dynamics as shown in Wu et al. [20], where a sinusoidal wave passes via a ship or offshore structure, with the accompanying fluid actions varying with time $t$. In the aforementioned study [20], the differential equation is defined as

$$y^{(4)} + 3y'' + y(2 + \varepsilon \cos(\Omega t)) = 0, \quad t > 0,$$

with

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

where $\varepsilon = 0$ with regard to the theoretical solution, $y(t) = 2 \cos(t) - \cos(\sqrt{2} t)$. On the other hand, the

### Table 4: Numerical findings to solve Problem 2 with step size $h = (1/320)$.

<table>
<thead>
<tr>
<th>$t$</th>
<th>ZSDM6 [8]</th>
<th>IHB8 [12]</th>
<th>I2PBDO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.103125</td>
<td>0.666133814775 – 14</td>
<td>4.261834 – 14</td>
<td>0.0000000 + 00</td>
</tr>
<tr>
<td>0.206250</td>
<td>0.974775815621 – 13</td>
<td>7.314820 – 13</td>
<td>3.9099252 – 16</td>
</tr>
<tr>
<td>0.306250</td>
<td>0.461852778244 – 12</td>
<td>1.625321 – 12</td>
<td>7.2449456 – 16</td>
</tr>
<tr>
<td>0.406250</td>
<td>0.142241773915 – 11</td>
<td>6.568972 – 12</td>
<td>9.9936929 – 16</td>
</tr>
<tr>
<td>0.506250</td>
<td>0.342548212088 – 11</td>
<td>2.849147 – 12</td>
<td>1.2947886 – 15</td>
</tr>
<tr>
<td>0.603125</td>
<td>0.687361279006 – 11</td>
<td>4.231457 – 12</td>
<td>0.0000000 + 00</td>
</tr>
<tr>
<td>0.703125</td>
<td>0.126365584663 – 10</td>
<td>1.246852 – 11</td>
<td>1.5163355 – 15</td>
</tr>
<tr>
<td>0.803125</td>
<td>0.213882245248 – 10</td>
<td>5.146142 – 11</td>
<td>2.8632323 – 15</td>
</tr>
<tr>
<td>0.903125</td>
<td>0.339896999435 – 10</td>
<td>3.765423 – 11</td>
<td>4.5622733 – 15</td>
</tr>
</tbody>
</table>

### Table 5: Numerical findings to solve Problem 3.

<table>
<thead>
<tr>
<th>NS</th>
<th>FDM6</th>
<th>BHC8</th>
<th>FSHB8</th>
<th>I2PBDO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>1.25612 – 04</td>
<td>1.74 – 08</td>
<td>8.07 – 10</td>
<td>1.851538 – 10</td>
</tr>
<tr>
<td>40</td>
<td>1.90752 – 06</td>
<td>8.45 – 11</td>
<td>3.22 – 12</td>
<td>2.328543 – 12</td>
</tr>
<tr>
<td>80</td>
<td>2.96411 – 08</td>
<td>3.69 – 13</td>
<td>—</td>
<td>2.115566 – 14</td>
</tr>
<tr>
<td>100</td>
<td>8.65117 – 09</td>
<td>7.11 – 14</td>
<td>—</td>
<td>1.079081 – 14</td>
</tr>
</tbody>
</table>

### Table 6: Numerical findings to solve Problem 4, $t \in [0, 1]$.

<table>
<thead>
<tr>
<th>NS</th>
<th>SSB8</th>
<th>I2PBDO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>54</td>
<td>3.547029 – 11</td>
<td>1.455049 – 14</td>
</tr>
<tr>
<td>504</td>
<td>1.141416 – 10</td>
<td>6.226973 – 15</td>
</tr>
<tr>
<td>5004</td>
<td>1.439730 – 09</td>
<td>4.668800 – 14</td>
</tr>
<tr>
<td>50004</td>
<td>2.466095 – 09</td>
<td>4.156048 – 13</td>
</tr>
</tbody>
</table>

### Table 7: Numerical findings to solve Problem 5.

<table>
<thead>
<tr>
<th>NS</th>
<th>LBO7 [14]</th>
<th>I2PBDO6</th>
</tr>
</thead>
<tbody>
<tr>
<td>83</td>
<td>2.61 – 13</td>
<td>2.727680 – 14</td>
</tr>
<tr>
<td>145</td>
<td>2.84 – 14</td>
<td>4.776630 – 14</td>
</tr>
<tr>
<td>234</td>
<td>3.02 – 14</td>
<td>3.818367 – 15</td>
</tr>
</tbody>
</table>

### Table 8: Numerical findings to solve Problem 6.

<table>
<thead>
<tr>
<th>$h$</th>
<th>FCN</th>
<th>Method</th>
<th>MAXE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>404</td>
<td>RKT5</td>
<td>5.327998 – 5</td>
</tr>
<tr>
<td>2424</td>
<td>RKF5</td>
<td>2.064967 – 3</td>
<td></td>
</tr>
<tr>
<td>0.025</td>
<td>2800</td>
<td>I2PBDO6</td>
<td>1.691679 – 13</td>
</tr>
<tr>
<td>1600</td>
<td>RKT5</td>
<td>4.254471 – 7</td>
<td></td>
</tr>
<tr>
<td>9600</td>
<td>RKF5</td>
<td>1.917218 – 6</td>
<td></td>
</tr>
<tr>
<td>0.0125</td>
<td>5600</td>
<td>I2PBDO6</td>
<td>8.014250 – 14</td>
</tr>
<tr>
<td>3204</td>
<td>RKT5</td>
<td>1.786611 – 8</td>
<td></td>
</tr>
<tr>
<td>19224</td>
<td>RKF5</td>
<td>6.053233 – 8</td>
<td></td>
</tr>
<tr>
<td>0.00625</td>
<td>11200</td>
<td>I2PBDO8</td>
<td>3.008073 – 14</td>
</tr>
<tr>
<td>6404</td>
<td>RKT5</td>
<td>6.002665 – 9</td>
<td></td>
</tr>
<tr>
<td>38424</td>
<td>RKF5</td>
<td>4.878530 – 9</td>
<td></td>
</tr>
</tbody>
</table>

### Table 9: Numerical findings to solve Problem 7.

<table>
<thead>
<tr>
<th>$h$</th>
<th>FCN</th>
<th>Method</th>
<th>MAXE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>210</td>
<td>I2PBDO6</td>
<td>1.375600 – 3</td>
</tr>
<tr>
<td>720</td>
<td>RKF5</td>
<td>1.534759 – 1</td>
<td></td>
</tr>
<tr>
<td>0.025</td>
<td>840</td>
<td>I2PBDO6</td>
<td>4.992115 – 7</td>
</tr>
<tr>
<td>484</td>
<td>RKT5</td>
<td>3.920592 – 5</td>
<td></td>
</tr>
<tr>
<td>2904</td>
<td>RKF5</td>
<td>6.983629 – 4</td>
<td></td>
</tr>
<tr>
<td>0.0125</td>
<td>1680</td>
<td>I2PBDO6</td>
<td>8.294011 – 9</td>
</tr>
<tr>
<td>960</td>
<td>RKT5</td>
<td>2.560210 – 6</td>
<td></td>
</tr>
<tr>
<td>5760</td>
<td>RKF5</td>
<td>1.960240 – 5</td>
<td></td>
</tr>
<tr>
<td>0.00625</td>
<td>3360</td>
<td>I2PBDO8</td>
<td>1.330643 – 10</td>
</tr>
<tr>
<td>1920</td>
<td>RKT5</td>
<td>1.468912 – 7</td>
<td></td>
</tr>
<tr>
<td>11520</td>
<td>RKF5</td>
<td>4.878530 – 9</td>
<td></td>
</tr>
</tbody>
</table>
theoretical solution can be seen to be indeterminate when \( \epsilon \neq 0 \) as can be seen in [21]. Numerical investigations to resolve the ordinary differential equations of the fourth-order have been expanded for resolving a problem from ship dynamics. Numerical consideration was offered when \( \epsilon \leq 0 \) and \( \epsilon \leq 1 \) with \( \Omega = 0.25(\sqrt{2} - 1) \) as can be seen in Twizell [21] and Cortell [22]. However, Twizell [21] and Cortell [22] derived the ODE method by lowering the equations to system of first-order (ODEs) rather than directly resolving it. Moreover, Cortell [22] introduced the extension of the classical RK method. Figure 3 depicts the exact and approximate solutions by using I2PBDO6 for the application of ship dynamics problem with \( \epsilon = 0 \). However, the results of equation (52), \( \epsilon = 1 \), and \( \Omega = 0.25(\sqrt{2} - 1) \) in [0,15] are shown in 4. It is obtained by I2PBDO6 with the solution obtained by Mathematica in-built package NDSolve, where \( \hat{h} = 0.1 \). From Figures 3 and 4, it is clear that the solutions that are obtained by I2PBDO6 strongly agree with not only the exact solutions but also with those of Mathematica in-built package NDSolve, respectively.

5. Conclusions

In this article, we proposed a two-point block method with extra derivative to directly solve initial value problems of the fourth order. The derivation of the new method can be easily implemented. This implicit block method is found to satisfy the property of convergence as suggested by the significant enhancement with regard to accuracy in the numerical results. Therefore, the proposed method I2PBDO6 is suitable for attaining high precision when solving fourth-order ODEs.

Data Availability

The data used to support the findings of this study are included within the article.
Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to thank the Universiti Putra Malaysia for the financial support through Putra grant (project no. GP-IPS/2020/9688400).

References


