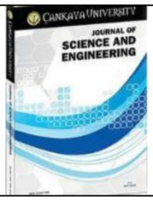
e-ISSN: 2564-7954 CUSJE 21(2): 113-132 (2024) Research Article



Çankaya University

Journal of Science and Engineering

CUJSE

https://dergipark.org.tr/cankujse

An Order (K+5) Block Hybrid Backward Differentiation Formula for Solution of Fourth Order Ordinary Differential Equations

Raihanatu Muhammad[1], Hajara Hussaini[1], Abdulmalik Oyedeji[2*]

[1] Department of Mathematics, Federal University of Technology Minna, Nigeria
[2] Department of Mathematics, University of Tennessee, Knoxville, U.S.A.

Keywords

Backward Differentiation Formula,
Block Method,
Interpolation Methods,
Linear Multistep Methods,
Collocation Method.

Abstract

This paper covers the derivation and implementation of the 4-step linear Multistep method of Block Hybrid Backward Differentiation Formula (BHBDF) for solving fourth-order initial value problems in ordinary differential equations. In the derivation of the proposed numerical method, the utilization of collocation and interpolation points was adopted with Legendre polynomials serving as the fundamental basis function. The 4-step BHBDF developed to solve fourth-order IVPs has a higher order of accuracy (p=9). Furthermore, the proposed numerical block methods are employed directly to solve fourth-order ODEs. In comparison to some existing methods examined in the prior studies, the proposed method has a robust implementation strategy and demonstrate a higher level of accuracy.

1. Introduction

The differential equation provides a powerful tool for modeling and analyzing dynamic systems, making them invaluable in the various fields of science and engineering, their understanding and applications are crucial for gaining insights into the behavior of complex phenomena [1]. The study of differential equations has a long history, dating back to the early days of calculus in the 17th century. The development of differential equations can be traced to the works of Isaac Newton, Gottfried Leibniz, and other mathematicians of the time.

Newton used differential equations to describe the motion of objects under the influence of gravity and other forces. A differential equation is a mathematical equation that describes how a function changes concerning one or more variables. It relates an unknown function and its derivatives to one or more independent variables [2]. The function and its derivatives are typically represented by the letter while the independent variables are represented by u. There are two types of differential equations: Ordinary differential equations (ODES) and Partial differential equations (PDEs). An ordinary differential equation (ODE) is a differential equation that involves only one independent variable; typically, time [3].

$$\frac{dy}{dx} = f(x,y) \tag{1}$$

Equation (1) is an example of a first-order ordinary differential equation (ODE) where y is an unknown function to be calculated and x is the independent variable, $f(x,y)$ is any function in terms of x, and y. The equation describes the rate of change of the variable y with respect to x, where $f(x,y)$ is a function that defines how y changes depending on both x and y. $\frac{dy}{dx}$ is the derivative of y with respect to x, representing the rate of change of

* Corresponding Author: aoyedeji@vols.utk.edu
Received: September 3, 2024, Accepted: October 7, 2024

y as x changes. $f(x, y)$is the function of two variables x and y that defines their relationship. It tells how the rate of change of y (i.e. $\frac{dy}{dx}$) depends on the current values of x and y .

Ordinary Differential Equations (ODEs) play a significant role in modeling real-life phenomena across various fields such as engineering, physics, biology, and economics. In particular, fourth-order ODEs arise in complex systems such as beam deflection analysis, fluid dynamics, quantum mechanics, and other critical applications where higher-order derivatives dictate the system's behavior. Due to their complexity, developing efficient and accurate numerical methods to solve these equations is of utmost importance.

The Backward Differentiation Formula (BDF) is one of the most widely used numerical techniques for solving stiff ODEs. Known for its stability and effectiveness in addressing systems with rapid variations, BDF is particularly well-suited for long-term integration of stiff equations. However, when dealing with higher-order ODEs, the traditional BDF methods face challenges in terms of accuracy, convergence, and computational efficiency.

In this research, we propose the development of an innovative numerical scheme termed the Order (K + 5) Block Hybrid Backward Differentiation Formula (BHBDF). This method is tailored specifically for the solution of fourth-order ODEs. The incorporation of a block hybrid structure allows the method to solve multiple points of the solution grid simultaneously, thereby enhancing computational efficiency. Moreover, the extension to order K+5 improves accuracy while maintaining the stability characteristics needed for solving stiff problems.

The introduction of this higher-order block hybrid method aims to bridge the gap between accuracy and efficiency in solving higher-order ODEs, with a particular focus on applications in real-world problems. The proposed method will be rigorously analyzed for stability, consistency, and convergence, and its performance will be benchmarked against existing methods. This work is not only of mathematical significance but also has broad applicability in solving real-life problems across multiple disciplines, including areas aligned with the United Nations Sustainable Development Goals (SDGs). By improving the numerical methods available for solving fourth-order ODEs, this research has the potential to contribute to advancements in energy-efficient technologies, environmental modeling, and sustainable industrial processes.

The motivation for this research is the desire to obtain more accurate and efficient results; thus, the research will focus on a block approach of the backward differentiation formula (BDF) that generates approximate solutions to ordinary differential equations across the entire integration interval.

2. Methodology

2.1. Derivation of the Numerical Schemes

In this section we present the derivation of a hybrid backward differentiation formula, employing a class block approach, for the solution of fourth-order ordinary differential equations (ODEs). The derivation process begins with an orthogonal polynomial of the form:

$$U(t) = \sum_{i=0}^{r+s-1} c_i L_i(t) \tag{2}$$

The main properties of equation (2) is that it’s a function that is a linear combination of basis functions, with its accuracy determined by the degree of these functions. Its smoothness is inherited from the smoothness of the basis functions, and it can interpolate specific points when appropriate basis functions like Lagrange polynomials are used. The behavior of U(t) depends on the chosen basis, and the coefficients c_i shape the function based on the problem's requirements. If the basis functions are orthogonal, computations are simplified, and as more basis functions are included, U(t) may converge to the exact solution of a given problem.

Also, equation (2) has a flexible representation that inherits key properties from the chosen basis functions, such as smoothness, degree of approximation, and interpolation behavior. It provides a foundation for constructing solutions to complex problems in numerical analysis, particularly in solving differential equations, interpolation, and approximation theory.

where **'r'** represents the number of interpolation points and **'s'** denotes the number of collocation points, the coefficients $c_i's$ mentioned are the coefficients of the Legendre polynomial $L_i(t)$, which will be determined through the derivation process. This paper proposes a class of 4-step block hybrid methods for solving fourth-order ordinary differential equations (ODEs).

4-Step Block Hybrid BDF for Fourth Order ODEs (BHBDF IV)

The degree of the Legendre polynomial used in the proposed method to obtain an approximate to exact solution of ODEs is 12. The polynomial is provided below:

$$\left.\begin{aligned}
U(t) &= c_0 + c_1 t + \left(\frac{3}{2}t^2 - \frac{1}{2}\right)c_2 + \left(\frac{5}{2}t^3 - \frac{3}{2}t\right)c_3 + \left(\frac{35}{8}t^4 - \frac{15}{4}t^2 + \frac{3}{8}\right)c_4 \\
&+ \left(\frac{63}{8}t^5 - \frac{35}{4}t^3 + \frac{15}{8}t\right)c_5 + \left(\frac{231}{16}t^6 - \frac{315}{16}t^4 + \frac{105}{16}t^2 - \frac{5}{16}\right)c_6 \\
&+ \left(\frac{429}{16}t^7 - \frac{693}{16}t^5 + \frac{315}{16}t^3 - \frac{35}{16}t\right)c_7 + \left(\frac{6435}{128}t^8 - \frac{3003}{32}t^6 + \frac{3465}{64}t^4 - \frac{315}{32}t^2 + \frac{35}{128}\right)c_8 \\
&+ \left(\frac{12155}{128}t^9 - \frac{6435}{32}t^7 + \frac{9009}{64}t^5 - \frac{1155}{32}t^3 + \frac{315}{128}t\right)c_9 \\
&+ \left(\frac{46189}{256}t^{10} - \frac{109395}{256}t^8 + \frac{45045}{128}t^6 - \frac{15015}{256}t^4 + \frac{3465}{256}t^2 - \frac{63}{256}\right)c_{10} \\
&+ \left(\frac{88179}{256}t^{11} - \frac{230935}{256}t^9 + \frac{109395}{128}t^7 - \frac{45045}{128}t^5 + \frac{15015}{256}t^3 - \frac{693}{256}t\right)c_{11} \\
&+ \left(\frac{676039}{512}t^{12} - \frac{969969}{512}t^{10} + \frac{2078505}{1024}t^8 - \frac{225225}{1024}t^6 + \frac{225225}{1024}t^4 - \frac{9009}{512}t^2 + \frac{231}{1024}\right)c_{12}
\end{aligned}\right\} \quad (3)$$

Equation (3) can be rewritten as $U(t) = \sum_{i=0}^{12} P_i(t)c_i$

Where $P_i(t)$ represents the polynomial associated with each coefficient c_i.

Breaking it down:

$$P_0(t) = 1$$

$$P_1(t) = t$$

$$P_2(t) = \frac{3}{2}t^2 - \frac{1}{2}$$

$$P_3(t) = \frac{5}{2}t^3 - \frac{3}{2}t$$

$$\vdots$$

$$P_{12}(t) = \frac{676039}{512}t^{12} - \frac{969969}{512}t^{10} + \frac{2078505}{1024}t^8 - \frac{255225}{1024}t^6 + \frac{255225}{1024}t^4 - \frac{9009}{512}t^2 + \frac{231}{1024}$$

Equation (3) was interpolated at $t = \frac{t_{i+j}}{5}, 0 \leq j < 4, \left(t_{i+j} = t_i + jh\right)$ and its second derivative collocated at $t = t_{i+4}$.

The matrix inversion technique is employed using the Maple software to solve for the unknown $c_j's$. The values $c_j's$ are obtained, they are substituted back into equation (3), which is then simplified to obtain the continuous collocation method of the form:

$$U(t) = \alpha_0(t)u_i + \alpha_{\frac{1}{5}}(t)u_{i+\frac{1}{5}} + \alpha_{\frac{2}{5}}(t)u_{i+\frac{2}{5}} + \alpha_1(t)u_{i+1} + \alpha_{\frac{6}{5}}(t)u_{i+\frac{6}{5}} + \alpha_{\frac{7}{5}}(t)u_{i+\frac{7}{5}} + \alpha_2(t)u_{i+2} + \alpha_{\frac{11}{5}}(t)u_{i+\frac{11}{5}} + \alpha_{\frac{12}{5}}(t)u_{i+\frac{12}{5}} + \alpha_3(t)u_{i+3} + \alpha_{\frac{16}{5}}(t)u_{i+\frac{16}{5}} + \alpha_{\frac{17}{5}}(t)u_{i+\frac{17}{5}} + h^4\beta_4(t)v_{i+4} \quad (4)$$

Equation (4) signifies the continuous scheme of the proposed method, which, following the tradition of BDF, has only one evaluation point $t = t_i + 4h$. Upon implementation, the scheme resulting from equation (4) is given below.

$$u_{i+4} = -\frac{21902734334871}{6894678641246}u_i + \frac{4017668323500}{202784665919}u_{i+\frac{1}{5}} - \frac{683562244750}{18434969629}u_{i+\frac{2}{5}} + \frac{74638822252176}{202784665919}u_{i+1} - \frac{197532942340500}{202784665919}u_{i+\frac{6}{5}} + \frac{16138811223000}{18434969629}u_{i+\frac{7}{5}} - \frac{25752063708774}{18434969629}u_{i+2} + \frac{433136219698000}{202784665919}u_{i+\frac{11}{5}} - \frac{227165447588625}{202784665919}u_{i+\frac{12}{5}} + \frac{6272521913016}{18434969629}u_{+3} - \frac{114602604377625}{405569331838}u_{i+\frac{16}{5}} + \frac{260460154714500}{3447339320623}u_{i+\frac{17}{5}} + \frac{13444704}{18434969629}h^4v_{i+4} \quad (5)$$

To solve ODEs using the proposed block method, it is necessary to implement the method. This requires the generation of additional schemes from the same continuous scheme. To achieve this, the second derivative of equation (4) is evaluated.

$$t_{i+j}; j = \frac{1}{5}, \frac{2}{5}, \frac{6}{5}, \frac{7}{5}, 2, \frac{11}{5}, \frac{12}{5}, 3, \frac{16}{5}, \frac{17}{5}, 4$$

Some of the schemes obtained are presented below:

$$u_{i+\frac{1}{5}} = \frac{4218137605185326997}{18300379968770950 39}u_i + \frac{1428197888590065 56}{107649293933946767}u_{i+\frac{2}{5}} - \frac{582764423427187548}{107649293933946767}u_{i+1} + \frac{1200512813135943776}{107649293933946767}u_{i+\frac{6}{5}} - \frac{854212220926047368}{107649293933946767}u_{i+\frac{7}{5}} + \frac{714267669417287348}{107649293933946767}u_{i+2} - \frac{883538302946584376}{107649293933946767}u_{i+\frac{11}{5}} + \frac{372847337842375068}{107649293933946767}u_{i+\frac{12}{5}} - \frac{53895368868755216}{107649293933946767}u_{i+3} + \frac{32842635714989867}{107649293933946767}u_{+\frac{16}{5}} - \frac{102734553258915477}{1830037996877095039}u_{i+\frac{17}{5}} - \frac{1226441659478112}{10226682923724942865}h^4v_{i+\frac{1}{5}} + \frac{6455625577632}{10226682923724942865}h^4v_{i+4} \quad (6)$$

$$\begin{aligned} u_{i+\frac{2}{5}} = & -\frac{138050321353493991}{293959033656356266}u_i + \frac{202157832352066526}{146979516828178133}u_{i+\frac{1}{5}} - \frac{443160563131969176}{146979516828178133}u_{i+1} \\ & + \frac{1290415628659943422}{146979516828178133}u_{i+\frac{6}{5}} - \frac{99932003552265716}{13361774257107103}u_{i+\frac{7}{5}} + \frac{103937516693660741}{13361774257107103}u_{i+2} \\ & - \frac{1462910454365395672}{146979516828178133}u_{i+\frac{11}{5}} + \frac{633365830923358026}{146979516828178133}u_{i+\frac{12}{5}} - \frac{8741569495941572}{13361774257107103}u_{i+3} \\ & + \frac{118459428884418323}{293959033656356266}u_{+\frac{16}{5}} - \frac{10996691475275142}{146979516828178133}u_{i+\frac{17}{5}} + \frac{80523947339472}{66808871285535515}h^4 v_{i+\frac{2}{5}} \\ & + \frac{58767163728}{66808871285535515}h^4 v_{i+4} \end{aligned} \tag{7}$$

$$\begin{aligned} u_{i+1} = & \frac{145532389975992 43}{99042461351078784}u_i - \frac{48061248425}{139085827404}u_{i+\frac{1}{5}} + \frac{1104595557175}{3616231512504}u_{i+\frac{2}{5}} \\ & + \frac{855104438825}{417257482212}u_{i+\frac{6}{5}} - \frac{710064253525}{486800395914}u_{i+\frac{7}{5}} + \frac{341855274079}{278171654808}u_{i+2} \\ & - \frac{582960851725}{382486025361}u_{i+\frac{11}{5}} + \frac{179342777675}{278171654808}u_{i+\frac{12}{5}} - \frac{254258496217}{2712173634378}u_{i+3} \\ & + \frac{254661727225}{4450746476928}u_{+\frac{16}{5}} - \frac{141264240025}{13398601373252}u_{i+\frac{17}{5}} + \frac{18434969629}{1576306043912}hu'_i \\ & + \frac{1386}{11590485617}h^4 v_{i+4} \end{aligned} \tag{8}$$

$$\begin{aligned} u_{i+2} = & -\frac{857872900697955311}{1761806337981579837 2}u_i + \frac{456840638182067 50}{259089167350232329}u_{i+\frac{1}{5}} - \frac{634982906921498 75}{306196288686638207}u_{i+\frac{2}{5}} \\ & + \frac{202108128452339880}{259089167350232329}u_{i+1} - \frac{420151839483570250}{259089167350232329}u_{i+\frac{6}{5}} + \frac{27461383121085500}{23553560668202939}u_{i+\frac{7}{5}} \\ & + \frac{322524395263189000}{259089167350232329}u_{i+\frac{11}{5}} - \frac{136851371595198625}{259089167350232329}u_{i+\frac{12}{5}} + \frac{23674548638835708}{306196288686638207}u_{i+3} \\ & - \frac{48985463478380125}{1036356669400929316}u_{+\frac{16}{5}} + \frac{38415459147188250}{4404515844953949593}u_{i+\frac{17}{5}} + \frac{9291224693016}{23553560668202939}h^2 u''_i \\ & - \frac{2348217648}{23553560668202939}h^4 v_{i+4} \end{aligned} \tag{9}$$

$$\begin{aligned} u_{i+3} = & \frac{2634120195610189978 77}{7506320971812443322 52}u_i - \frac{3377244346148947462 5}{2207741462297777447 8}u_{i+\frac{1}{5}} + \frac{830331413175524362 5}{401407538599595899 6}u_{i+\frac{2}{5}} \\ & - \frac{10254627061408465234 2}{1103870731148888723 9}u_{i+1} + \frac{4352752765745309038 75}{2207741462297777447 8}u_{i+\frac{6}{5}} - \frac{1443716901357932662 5}{100351884649898974 9}u_{i+\frac{7}{5}} \\ & + \frac{5086727688403471761 3}{401407538599595899 6}u_{i+2} - \frac{1751158346049541747 50}{1103870731148888723 9}u_{i+\frac{11}{5}} + \frac{7465140158075543287 5}{1103870731148888723 9}u_{i+\frac{12}{5}} \\ & + \frac{2705994760334551237 5}{4415482924595554895 6}u_{+\frac{16}{5}} - \frac{4253280393608629987 5}{3753160485906221661 26}u_{i+\frac{17}{5}} + \frac{362357763027624}{100351884649898974 9}h^3 u'''_i \\ & + \frac{1315591677036}{100351884649898974 9}h^4 v_{i+4} \end{aligned} \tag{10}$$

2.2. Convergence Properties of the Methods

Within this section, we investigate the consistency, order, and error constant to assess the zero stability. According to [4], the zero stability and consistency of a linear multi-step method are both necessary and sufficient conditions for the method to converge. To establish convergence of the methods, we look at it in two phases.

2.2.1. Consistency

Theorem 1: The necessary and sufficient condition for a Linear Multistep Method to be convergent is that it is consistent [5].

Proof: We are required to show that $\sum_{j=0}^{k} \boldsymbol{\alpha_j} = \mathbf{0}$ and $\sum_{j=0}^{k} \boldsymbol{\beta_j} = \sum_{j=0}^{k} \boldsymbol{j\,\alpha_j}$

Let $\boldsymbol{t} = \boldsymbol{t}^*$ be a fixed point. Assume that the $\boldsymbol{k}$-step method is converging to a function $\boldsymbol{y}(\boldsymbol{t})$ as $\boldsymbol{h} \to \mathbf{0}$. Then,
$\boldsymbol{y_{n+j}} \to \boldsymbol{y}(\boldsymbol{t}^*) \quad \forall \quad \boldsymbol{j} = \mathbf{0}, \mathbf{1}, \mathbf{2} \dots, \boldsymbol{k}$

Hence, we write
$\boldsymbol{y}(\boldsymbol{t}^*) = \boldsymbol{y_{n+j}} + \boldsymbol{\theta_{j,n}}(\boldsymbol{h}) \quad \forall \quad \boldsymbol{j} = \mathbf{0}, \mathbf{1}, \mathbf{2}, \dots, \boldsymbol{k}$ where $\lim_{\boldsymbol{h}\to\mathbf{0}} \boldsymbol{\theta_{j,n}}(\boldsymbol{h}) = \mathbf{0}$

Therefore

$$\sum_{j=0}^{k} \alpha_j y(t^*) = \sum_{j=0}^{k} \alpha_j\, y_{n+j} + \sum_{j=0}^{k} \alpha_j\, \theta_{j,n}(h)$$

$$\Rightarrow\; y(t^*) \sum_{j=0}^{k} \alpha_j = h \sum_{j=0}^{k} \beta_j\, f_{n+j} + \sum_{j=0}^{k} \alpha_j\, \theta_{j,n}(h)$$

taking limits as $h \to 0$, the terms

$h \sum_{j=0}^{k} \beta_j\, f_{n+j} \to 0$ and $\sum_{j=0}^{k} \alpha_j\, \theta_{j,n}(h) \to 0$

and since $y(t^*)$ will not necessary be zero $\Rightarrow \sum_{j=0}^{k} \alpha_j = 0$ (i)

Therefore, we are done with the first condition. In fact, (i) is equivalent to the condition $C_0 = 0$ (details in the next subsection) We conclude that (1) is satisfied.

To prove the second condition, note that we have not said anything about the function $y' = f$. Suppose that $\frac{y_{n+j} - y_n}{jh} \to y'(t^*) \quad \forall\, j = 1,2, \dots\dots, k$
$\Rightarrow y_{n+j} - y_n = jhy'(t^*) + jh\varphi_{j,n}(h) \quad \forall\; j = 1, 2, \dots., k$
where $\varphi_{j,n}(h) \to 0$ as $h \to 0$

Therefore

$$\sum_{j=0}^{k} \alpha_j\, y_{n+j} - \sum_{j=0}^{k} \alpha_j\, y_n = h \sum_{j=0}^{k} j\, \alpha_j y'(t^*) + h \sum_{j=0}^{k} j\alpha_j\, \varphi_{j,n}(h)$$

$\Rightarrow$

$$h \sum_{j=0}^{k} \beta_j\, f_{n+j} - y_n \sum_{j=0}^{k} \alpha_j = hy'(t^*) \sum_{j=0}^{k} j\, \alpha_j + h \sum_{j=0}^{k} j\alpha_j\, \varphi_{j,n}(h)$$

Given that (i) holds, we divide through by h we have

$$\sum_{j=0}^{k} \beta_j f_{n+j} = y'(t^*) \sum_{j=0}^{k} j\,\alpha_j + \sum_{j=0}^{k} j\alpha_j\,\varphi_{j,n}(h)$$

If $y_{n+j} \to y(t^*)$ as $h \to 0\ \forall\, j \in \{0,1,2, \ldots\ldots, k\}$ then $f_{n+j} \to f(t^*, y(t^*))$.

Hence letting $h \to 0$ we get

$$f\big(t^*, y(t^*)\big) \sum_{j=0}^{k} \beta_j = y'(t^*) \sum_{j=0}^{k} j\,\alpha_j$$

Hence $\quad y'(t^*) = f\big(t^*, y(t^*)\big) \quad$ if $\quad \sum_{j=0}^{k} \beta_j = \sum_{j=0}^{k} j\,\alpha_j \qquad (ii)$

Condition (ii) is equivalent to $C_1 = 0$ which gives rise to the definition that “A linear multistep method that is at least first order is called a consistent method”.

From the above analysis, we conclude that a necessary condition for a k-step linear multistep method to be convergent is that the method is consistent.

2.2.2. Zero stability

Theorem 2: The necessary and sufficient conditions for a Linear Multistep Method to be convergent is that it is zero-stable [6].

Proof: To prove zero stability, we first define the first and second characteristic polynomials of a general k −step method as

$$\rho(\xi) = \sum_{j=0}^{k} \alpha_j \xi^j = \alpha_k \xi^k + \alpha_{k-1}\xi^{k-1} + \alpha_{k-2}\xi^{k-2} + \cdots + \alpha_0 \qquad (i)$$

and

$$\sigma(\xi) = \sum_{j=0}^{k} \beta_j \xi^j = \beta_k \xi^k + \beta_{k-1}\xi^{k-1} + \beta_{k-2}\xi^{k-2} + \cdots + \beta_0 \qquad (ii)$$

respectively. The conditions that a consistent method must satisfy can be stated in terms of these polynomials. For example,

$$C_0 = 0 \iff \sum_{j=0}^{k} \alpha_j = 0 \Leftrightarrow \rho(1) = 0 \qquad (iii)$$

and

$$C_1 = 0 \iff \sum_{j=0}^{k} (j\alpha_j - \beta_j) = 0 \Leftrightarrow \rho'(1) - \sigma(1) = 0 \qquad (iv)$$

The condition (iii) shows us that $\xi = 1$ is a solution of the equation

$$\alpha_k \xi^k + \alpha_{k-1}\xi^{k-1} + \alpha_{k-2}\xi^{k-2} + \cdots + \alpha_2\xi^2 + \alpha_1\xi + \alpha_0 = 0 \qquad (v)$$

The roots of this equation play an important role in the behavior of the method. For example, consider the trivial problem $y' = 0,\ y(0) = 0$ which has the exact solution $y(t) \equiv 0$. If the linear multistep method is applied to this example, it will be found that the numerical solution satisfies the k^{th} degree homogeneous difference equation

$$\sum_{j=0}^{k} \alpha_j \, y_{n+j} = 0 \qquad (vi)$$

Here we have different possibilities but for the sake of convergence, we consider the case where the roots of (v) are real and distinct. Then the solution of (vi) which has starting values $y_\mu \to y(0)$ as $h \to 0,\ \mu \in \{0,1,2, \dots , k-1\}$ is

$$y_n = h(d_1\xi_1^n + \ d_2\xi_2^n + \cdots + d_k\xi_k^n) \qquad (vii)$$

where the $d_i's$ are arbitrary constants. If the method is to be convergent then we must have $y_n \to y(0)$ as $h \to 0, n \to \infty$ and $nh = t_n$ is fixed. However,

$$\lim_{\substack{h\to 0\\ nh=t_n}} h\xi_n = \lim_{n\to\infty} \frac{t_n}{n}\xi_s^n = \ t_n \lim_{n\to\infty} \frac{1}{n}\xi_s^n = 0 \Leftrightarrow \ |\xi_s^n| \le 1$$

t_n is a constant and its fixed so that is why its factored out of the limit operation. the limit simplifies to 0 as $n \to \infty$.

The expression can be further explained thus:

As n→∞ the term $\frac{1}{n}\xi_s^n$ approaches 0.

The limit equals 0 if and only if $|\xi_s^n| \le 1$

The behavior of the sequence ξ_s^n determines whether the entire expression converges to 0.

Hence this motivates the definition that a linear multistep method is said to be zero stable if no root of the first characteristic polynomial $\rho(\xi)$ has a modulus greater than one, and every root with a modulus of one is simple.

Hence, we have shown that the necessary condition for convergence is that the method be zero-stable.

Thus, consistency and zero-stability are necessary conditions for convergence as required.

2.2.3. Order and Error Constants

The definition of the linear differential operator, denoted as L, which is associated with the method is as follows [7,8]:

$$L[u(t);h] = \sum_{j=0}^{k}\left[\alpha_j u(t+jh) - h^n \beta_j u^{(n)}(t+jh)\right] \qquad (11)$$

The variable *n* represents the order of the considered differential equation.

Expanding (11) in the Taylor series, we have

$$L[u(t);h] = c_0 u(t) + \ c_1 h u'(t) + c_2 h^2 u''(t) + \cdots + c_q h^q u^q(t) \qquad (12)$$

where

$$c_0 = \alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 + \ldots + \alpha_k$$

$$c_1 = \alpha_1 + 2\alpha_2 + 3\alpha_3 + \ldots + k\alpha_k$$
$$c_2 = \frac{1}{2!}(\alpha_1 + 2^2\alpha_2 + 3^2\alpha_3 + \ldots + k^2\alpha_k) - (\beta_0 + \beta_1 + \beta_2 + \ldots + \beta_k)$$
$$\vdots$$
$$c_p = \frac{1}{p!}(\alpha_1 + 2^p\alpha_2 + 3^p\alpha_3 + \ldots + k^p\alpha_k) - \frac{1}{(p-n)!}(\beta_1 + 2^{p-2}\beta_2 + 3^{p-2}\beta_3 + \ldots + k^{p-2}\beta_k)$$
$$p \geq n$$

Definition 2.1: A Linear Multistep Method is said to be of order p if co $c_0 = c_1 = c_2 = \ldots = c_p = c_{p+1} = 0$ and $c_{p+n} \neq 0$ is the error constant. n represents the step number used in the method

Definition 2.2: A Linear Multistep Method is said to be consistent if it has an order of accuracy $p \geq 1$

2.2.4. Order and Error Constants of BHBDF IV

Considering the discrete scheme (5), the coefficients are given as follows:

$$\alpha_0 = -\frac{21902734334871}{6894678641246}, \alpha_{\frac{1}{5}} = \frac{4017668323500}{202784665919}, \alpha_{\frac{2}{5}} = -\frac{683562244750}{18434969629},$$
$$\alpha_1 = \frac{74638822252176}{202784665919}$$
$$\alpha_{\frac{6}{5}} = -\frac{197532942340500}{202784665919}, \alpha_{\frac{7}{5}} = \frac{16138811223000}{18434969629}, \alpha_2 = -\frac{25752063708774}{18434969629},$$
$$\alpha_{\frac{11}{5}} = \frac{433136219698000}{202784665919}, \alpha_{\frac{12}{5}} = -\frac{227165447588625}{202784665919}, \alpha_3 = \frac{6272521913016}{18434969629},$$
$$\alpha_{\frac{16}{5}} = -\frac{114602604377625}{405569331838}, \alpha_{\frac{17}{5}} = \frac{260460154714500}{3447339320623}, \alpha_4 = -1, \beta_4 = -\frac{13444704}{18434969629}$$

and applying (12),

$$c_0 = \alpha_0 + \alpha_{\frac{1}{5}} + \alpha_{\frac{2}{5}} + \alpha_1 + \alpha_{\frac{6}{5}} + \alpha_{\frac{7}{5}} + \alpha_2 + \alpha_{\frac{11}{5}} + \alpha_{\frac{12}{5}} + \alpha_3 + \alpha_{\frac{16}{5}} + \alpha_{\frac{17}{5}} + \alpha_4 = 0$$
$$c_1 = \left(\frac{1}{5}\right)\alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)\alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)\alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)\alpha_{\frac{7}{5}} + (2)\alpha_2 + \left(\frac{11}{5}\right)\alpha_{\frac{11}{5}} + \left(\frac{12}{5}\right)\alpha_{\frac{12}{5}} + (3)\alpha_3 + \left(\frac{16}{5}\right)\alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)\alpha_{\frac{17}{5}} + (4)\alpha_4 = 0$$

$$c_2 = \frac{1}{2!}\left(\begin{array}{l} \left(\frac{1}{5}\right)^2\alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)^2\alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)^2\alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)^2\alpha_{\frac{7}{5}} + (2)^2\alpha_2 + \left(\frac{11}{5}\right)^2\alpha_{\frac{11}{5}} + \left(\frac{12}{5}\right)^2\alpha_{\frac{12}{5}} + (3)^2\alpha_3 \\ + \left(\frac{16}{5}\right)^2\alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)^2\alpha_{\frac{17}{5}} + (4)^2\alpha_4 \end{array}\right) = 0$$

$$c_3 = \frac{1}{3!}\left(\begin{array}{l} \left(\frac{1}{5}\right)^3\alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)^3\alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)^3\alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)^3\alpha_{\frac{7}{5}} + (2)^3\alpha_2 + \left(\frac{11}{5}\right)^3\alpha_{\frac{11}{5}} + \left(\frac{12}{5}\right)^3\alpha_{\frac{12}{5}} + (3)^3\alpha_3 \\ + \left(\frac{16}{5}\right)^3\alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)^3\alpha_{\frac{17}{5}} + (4)^3\alpha_4 \end{array}\right) = 0$$

$$c_4 = \frac{1}{4!}\left(\begin{array}{l}\left(\frac{1}{5}\right)^4 \alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)^4 \alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)^4 \alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)^4 \alpha_{\frac{7}{5}} + (2)^4\alpha_2 + \left(\frac{11}{5}\right)^4 \alpha_{\frac{11}{5}} + \left(\frac{12}{5}\right)^4 \alpha_{\frac{12}{5}} + (3)^4\alpha_3 \\ + \left(\frac{16}{5}\right)^4 \alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)^4 \alpha_{\frac{17}{5}} + (4)^4\alpha_4\end{array}\right) - (\beta_4) = 0$$

$$c_5 = \frac{1}{5!}\left(\begin{array}{l}\left(\frac{1}{5}\right)^5 \alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)^5 \alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)^5 \alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)^5 \alpha_{\frac{7}{5}} + (2)^5\alpha_2 + \left(\frac{11}{5}\right)^5 \alpha_{\frac{11}{5}} + \left(\frac{12}{5}\right)^5 \alpha_{\frac{12}{5}} + (3)^5\alpha_3 \\ + \left(\frac{16}{5}\right)^5 \alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)^5 \alpha_{\frac{17}{5}} + (4)^5\alpha_4\end{array}\right) - (4\beta_4) = 0$$

$$c_6 = \frac{1}{6!}\left(\begin{array}{l}\left(\frac{1}{5}\right)^6 \alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)^6 \alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)^6 \alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)^6 \alpha_{\frac{7}{5}} + (2)^6\alpha_2 + \left(\frac{11}{5}\right)^6 \alpha_{\frac{11}{5}} + \left(\frac{12}{5}\right)^6 \alpha_{\frac{12}{5}} + (3)^6\alpha_3 \\ + \left(\frac{16}{5}\right)^6 \alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)^6 \alpha_{\frac{17}{5}} + (4)^6\alpha_4\end{array}\right) - \frac{1}{2!}(4^2\beta_4) = 0$$

$$c_7 = \frac{1}{7!}\left(\begin{array}{l}\left(\frac{1}{5}\right)^7 \alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)^7 \alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)^7 \alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)^7 \alpha_{\frac{7}{5}} + (2)^7\alpha_2 + \left(\frac{11}{5}\right)^7 \alpha_{\frac{11}{5}} + \left(\frac{12}{5}\right)^7 \alpha_{\frac{12}{5}} + (3)^7\alpha_3 \\ + \left(\frac{16}{5}\right)^7 \alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)^7 \alpha_{\frac{17}{5}} + (4)^7\alpha_4\end{array}\right) - \frac{1}{3!}(4^3\beta_4) = 0$$

$$c_8 = \frac{1}{8!}\left(\begin{array}{l}\left(\frac{1}{5}\right)^8 \alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)^8 \alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)^8 \alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)^8 \alpha_{\frac{7}{5}} + (2)^8\alpha_2 + \left(\frac{11}{5}\right)^8 \alpha_{\frac{11}{5}} + \left(\frac{12}{5}\right)^8 \alpha_{\frac{12}{5}} + (3)^8\alpha_3 \\ + \left(\frac{16}{5}\right)^8 \alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)^8 \alpha_{\frac{17}{5}} + (4)^8\alpha_4\end{array}\right) - \frac{1}{4!}(4^4\beta_4) = 0$$

$$c_9 = \frac{1}{9!}\left(\begin{array}{l}\left(\frac{1}{5}\right)^9 \alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)^9 \alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)^9 \alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)^9 \alpha_{\frac{7}{5}} + (2)^9\alpha_2 + \left(\frac{11}{5}\right)^9 \alpha_{\frac{11}{5}} + \left(\frac{12}{5}\right)^9 \alpha_{\frac{12}{5}} + (3)^9\alpha_3 \\ + \left(\frac{16}{5}\right)^9 \alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)^9 \alpha_{\frac{17}{5}} + (4)^9\alpha_4\end{array}\right) - \frac{1}{5!}(4^5\beta_4) = 0$$

$$\vdots\ c_{13} = \frac{1}{13!}\left(\begin{array}{l}\left(\frac{1}{5}\right)^{13} \alpha_{\frac{1}{5}} + \left(\frac{2}{5}\right)^{13} \alpha_{\frac{2}{5}} + \alpha_1 + \left(\frac{6}{5}\right)^{13} \alpha_{\frac{6}{5}} + \left(\frac{7}{5}\right)^{13} \alpha_{\frac{7}{5}} + (2)^{13}\alpha_2 + \left(\frac{11}{5}\right)^{13} \alpha_{\frac{11}{5}} \\ + \left(\frac{12}{5}\right)^{13} \alpha_{\frac{12}{5}} + (3)^{13}\alpha_3 + \left(\frac{16}{5}\right)^{13} \alpha_{\frac{16}{5}} + \left(\frac{17}{5}\right)^{13} \alpha_{\frac{17}{5}} + (4)^{13}\alpha_4\end{array}\right)$$
$$-\frac{1}{9!}(4^9\beta_4) = \frac{170549518626}{396063800623046875}$$

Hence the method is of order p = 9 and the error constant is

$$c_{p+4} = \frac{170549518626}{396063800623046875}$$

A similar procedure is applied to other discrete schemes that constitute the block members of BHBDF IV and the summary of the order and error constants is given in the tables below:

Table 1. Order and Error Constants of BHBDF IV

Equation	**Order *p***	**Error constants c_{p+4}**
(6)	9	$4.306112257 \times 10^{-7}$
(7)	9	$2.161505806 \times 10^{-9}$
(8)	9	$3.241188074 \times 10^{-9}$
(9)	9	$4.186179775 \times 10^{-10}$
(10)	9	$1.638837784 \times 10^{-10}$
(11)	9	$9.557009794 \times 10^{-11}$
(12)	9	$-3.548793546 \times 10^{-10}$
(13)	9	$4.964264872 \times 10^{-11}$
(14)	9	$2.030559407 \times 10^{-10}$
(15)	9	$4.788549245 \times 10^{-10}$
(16)	9	$9.511847482 \times 10^{-10}$
(17)	9	$-8.184580835 \times 10^{-9}$

In this Table, the order p and error constants c_{p+4} for the Block Hybrid Backward Differentiation Formula (BHBDF) IV are presented for various equations. The table demonstrates the consistency of the method, as all equations have a uniform order of accuracy, $p = 9$. This high order indicates the precision of the method in approximating the solutions to the fourth-order ordinary differential equations (ODEs).

The table presents the error constants for different equations, all with an order $p = 9$. The error constants, denoted as c_{p+4} , provide insights into the precision of numerical methods related to each equation. The table demonstrates the consistency of the method, as all equations have a uniform order of accuracy, $p = 9$. This high order indicates the precision of the method in approximating the solutions to the fourth-order ordinary differential equations (ODEs).

The first column lists the equation numbers, from Equation (6) to Equation (17) while the second column shows that all the equations have an order of 9. This indicates that each equation belongs to a numerical method with a high level of accuracy, as a method's order determines how the error decreases as the step size is reduced. The third column contains the error constants c_{p+4}, which are key indicators of the accuracy of each method. Smaller error constants suggest that the method associated with that equation is more accurate, as the error introduced per step is lower.

The error constants range from very small positive values, such as $4.186179775 \times 10^{-10}$, to larger values like $9.511847482 \times 10^{-10}$. Most of the error constants are positive, except for a couple of instances where negative error constants are observed. For example, Equation (12) has a negative error constant of -3.548793546 $\times 10^{-10}$ and Equation (17) has a significantly larger negative error constant of -8.184580835 $\times 10^{-9}$. The smallest error constant is for Equation (11) at $9.557009794 \times 10^{-11}$ indicating that this equation's associated numerical scheme is highly accurate. Conversely, Equation (17) has the largest error constant in magnitude, -8.184580835$\times 10^{-9}$, which suggests that the method corresponding to this equation is less accurate compared to the others.

The consistency of the order (p = 9) across all equations indicates that they all belong to a class of methods with similar theoretical error behavior. However, the variations in the error constants reflect differences in the performance of these schemes. The positive and small error constants imply that most of the methods are quite precise, but the presence of negative error constants for certain equations, such as (12) and (17), indicates potential numerical instability or less favorable error behavior for these methods.

The table provides valuable insights into the accuracy of different equations within the same order (p = 9). Methods corresponding to equations with smaller positive error constants, such as (11), (9), and (10), are likely to perform better in practical applications, while those with larger or negative error constants, such as (12) and (17), may require further investigation to ensure their reliability.

3. Results and Discussion

3.1. Problem 3.1

Consider the nonlinear problem:

$$u^{(iv)} - \left(u'\right)^2 + uu''' = -4t^2 + e^t\left(1 - 4t + t^2\right)$$

with the following initial conditions

$u(0) = 1, u^{'}(0) = 1, u^{''}(0) = 3, u^{'''}(0) = 1$ and h = 0.031250

Exact Solution: $U(t) = t^2 + e^t$

Table 2 compares the exact solution, and the numerical solution derived from a new method at various points in time t. The values demonstrate how closely the numerical solution approximates the exact solution. The first column represents time (t), ranging from t=0.031250 to t=0.312500in increments of 0.03125. The second column provides the exact solution at each time point, which serves as the true or reference value. The third column presents the numerical solution generated by the new method at the same time points. At t=0.031250, the exact solution is 1.115170918075647624811708, while the numerical solution is 1.115170918075647622708296. The difference between the two is incredibly small, approximately 2.10×10^{-12}.10 indicating that the numerical method is extremely accurate at this point. As time increases, both the exact and numerical solutions increase, reflecting the growth of the function being modeled. For instance, at t=0.125000t, the exact solution is 1.651824697641270317824853, and the numerical solution is 1.651824697641270074999467, with a very small error of approximately 2.74×10^{-10}. At t = 0.312500, the exact solution is 3.718281828459045235360287, and the numerical solution is 3.718281828459037775872613. The error at this point remains small, approximately 7.46×10^{-9} showing that even for larger values of t, the proposed method maintains high accuracy. Across all time points, the difference between the exact and numerical solutions is minimal. The error remains very small, on the order of 10^{-9} or less, which implies that the new numerical method is highly precise. The fact that the errors are consistently small across the entire time interval suggests that the method does not lose accuracy as time increases, which is a positive attribute of the numerical approach. The function represented by the exact and numerical solutions shows an increasing trend over time. For example, from t = 0.031250 to t = 0.312500, the values increase from approximately 1.115 to 3.718, indicating a significant growth. The proposed method successfully tracks this growth with remarkable precision, ensuring that the numerical values closely follow the exact solutions at all points. The table demonstrates the high accuracy of the proposed method for approximating the exact solution over the given time range. The numerical solution remains extremely close to the exact solution, with errors remaining very small, even as the values of the function increase significantly. This high level of precision suggests that the new method is well-suited for applications requiring accurate numerical approximations of the exact solution over time. The table compares the exact solution of a differential equation with the numerical solution generated by the proposed method for different time steps. The exact and numerical solutions are nearly identical, with only minor differences in the decimal places, indicating the high accuracy of the new method. The numerical errors, although present, are extremely small (on the order of 10^{-12}) and grow slightly as time increases, suggesting the method is both accurate and stable for solving problems where precision is critical.

Table 2. Comparison of the Exact and Numerical Results for Problem 1

t	**Exact Solution**	**Numerical Solution (Proposed Method)**
0.031250	1.115170918075647624811708	1.115170918075647622708296
0.062500	1.261402758160169833921072	1.261402758160169813038702
0.093750	1.439858807576003103983744	1.439858807576003017841975
0.125000	1.651824697641270317824853	1.651824697641270074999467
0.156250	1.898721270700128146848651	1.898721270700127596853969
0.187500	2.182118800390508974875368	2.182118800390507893352858
0.218750	2.503752707470476521624549	2.503752707470474595012874
0.250000	2.865540928492467604579538	2.865540928492464415254034
0.281250	3.269603111156949663800127	3.269603111156944675207792
0.312500	3.718281828459045235360287	3.718281828459037775872613

The x-axis represents the time variable t which ranges from 0.031250 to 0.312500. The y - axis represents the values of both the exact and numerical solutions, which range from approximately 1.115 to 3.718. Both the exact solution and numerical solution curves increase steadily as time progresses. Starting at a value of approximately 1.115 at t = 0.031250, the values rise smoothly and significantly, reaching around 3.718 by t = 0.312500. The shape of the graph appeared as increasing smooth curve (exponential-like growth or a gradual rise). The exact solution curve represents the true or reference values and serves as the baseline. The numerical solution curve generated by the proposed method, will follow almost perfectly along the exact solution curve because the differences between the two solutions are extremely small (on the order of 10^{-9}or less). As a result, both curves are nearly indistinguishable indicating high accuracy of the proposed numerical method. Even though minor differences exist between the two solutions, they are so small that the numerical curve will visually overlap with the exact curve.

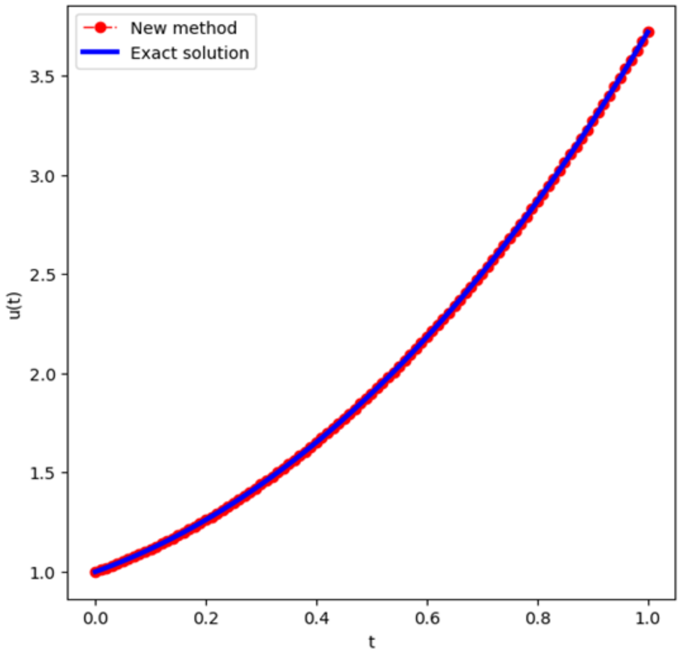


Figure 1. Comparison of exact and numerical solution of Problem 3.

The graph primarily shows one smooth curve for both the exact and numerical solutions, with almost no visible gap between them, indicating that the proposed method provides a nearly perfect approximation to the exact solution across the entire time range. The smooth increasing curve represents the growth of the function over time, and the close agreement between the exact and numerical curves highlights the effectiveness of the proposed method. The graph compares the exact solution, and the numerical solution obtained by the proposed method over a range of time steps (as seen in the table). The exact solution is the purple colored curve, representing the ideal or true values of the function over time. The numerical solution is the red line curve which follows the exact solution's curve.

The graph visually demonstrates that the proposed method produces highly accurate approximations of the exact solution, with minimal errors over time, and showcases the method's reliability and precision in solving differential equations.

Table 3 compares the error values across several methods, including those proposed [9-13] and the proposed method referred to as HBBDF IV. The errors are measured at different time steps t, showing how each method approximates the exact solution and the degree of error in doing so.

This method [9] generally produces moderate errors, which increase steadily as t increases. For example, at t = 0.031250, the error is 1.149 times 10^{-12}but by t = 0.312500, the error has grown to 1.396 times 10^{-18}, indicating that the method's accuracy degrades significantly over time.

[10,11] method shows much larger errors compared to the others. Starting from t = 0.031250 with an error of 1.788 times 10^{-10}, it quickly rises to 1.245 times 10^{-4} at t = 0.312500. This indicates that Kuboye's method struggles with maintaining accuracy as time progresses, resulting in the largest error of all the methods evaluated.

[12] method performs better than [9-11], particularly at smaller time steps. At t = 0.031250, the error is as small as 3.653 times $10^{-20}$and even at t = 0.312500, it only increases to 1.715\times 10^{-16}. This shows that Audu's method is highly accurate and maintains this accuracy well across increasing time steps.

This method [13] also performs well, but no error values are given for t from 0.187500 downwards. Up to that point, the error remains quite low, with values like 2.554 times 10^{-14} at t = 0.031250 and 7.023 times 10^{-13} at t = 0.125000 \). This indicates a moderate level of accuracy for earlier steps.

HBBDF IV (Proposed Method)**: The HBBDF IV method shows the highest level of accuracy across all time steps. Starting at t = 0.031250 with an error of 1.221 times 10^{-23}, the error remains impressively low throughout, only reaching 5.372 10^{-19} by t = 0.312500. This method outperforms the others by several orders of magnitude, demonstrating superior precision and stability.

[9-11] show significant error growth over time, indicating a reduction in accuracy for larger time steps. [12,13] maintain much better accuracy, with smaller errors throughout, though Abolarin's method does not provide values for later time steps.

The HBBDF IV method is clearly the most accurate, maintaining errors several orders of magnitude smaller than the other methods, making it the most precise and stable for this set of comparisons.

This detailed comparison highlights the superior performance of the HBBDF IV method in terms of both accuracy and error stability across all time steps.

Table 3. Comparison of Accuracy for Problem 3.1

t	**Error in [9]**	**Error in [10,11]**	**Error in [12]**	**Error in [13]**	**Error in HBBDF IV**
0.031250	1.149×10^{-12}	1.788×10^{-10}	3.653×10^{-20}	2.554×10^{-14}	1.221×10^{-23}
0.062500	1.885×10^{-11}	1.134×10^{-08}	2.827×10^{-19}	3.231×10^{-13}	5.425×10^{-22}
0.093750	9.780×10^{-11}	1.196×10^{-07}	1.005×10^{-18}	3.548×10^{-13}	3.390×10^{-21}
0.125000	3.166×10^{-10}	6.401×10^{-07}	3.228×10^{-18}	7.023×10^{-13}	1.176×10^{-20}
0.156250	7.909×10^{-10}	2.349×10^{-06}	8.360×10^{-18}	2.338×10^{-12}	3.028×10^{-20}
0.187500	1.676×10^{-09}	6.573×10^{-06}	1.858×10^{-17}	-	6.500×10^{-20}
0.218750	3.169×10^{-09}	1.610×10^{-05}	3.644×10^{-17}	-	1.234×10^{-19}
0.250000	5.512×10^{-09}	3.501×10^{-05}	6.529×10^{-17}	-	2.144×10^{-19}
0.281250	8.995×10^{-09}	6.985×10^{-05}	1.088×10^{-16}	-	3.483×10^{-19}
0.312500	1.396×10^{-08}	1.245×10^{-04}	1.715×10^{-16}	-	5.372×10^{-19}

3.2. Problem 3.2

Consider a sinusoidal wave of frequency **Ω** passes along a ship or offshore structure, the resultant fluid actions vary with time *t*. Therefore, consider the fourth-order problem as solved [14]:

$$u^{(iv)}+3u''+\Omega u\left(2+\varepsilon\cos wt\right)=0$$

with the following initial conditions

$u(0)=1, u'(0)=0, u''(0)=0, u'''(0)=0$ and h = 0.03125

where $\varepsilon=0$ for the existence of theoretical solution exact solution: $U(t)=2\,cos\,t-cos\,\sqrt{2}\,t$

Table 4. Comparison of the Exact and Numerical Results for Problem 3.2

t	**Exact Solution**	**Numerical Solution (Proposed Method)**
0.003125	0.999999999992052721790965 8	0.9999999999920527217876255
0.00625	0.9999999998728439211837312	0.9999999998728439211593905
0.00938	0.9999999993562754941897026	0.9999999993562754941109965
0.01250	0.9999999979655265806035874	0.9999999979655265804224698
0.01563	0.9999999950330675334441103	0.9999999950330675330875725
0.01875	0.9999999897006794757253819	0.9999999897006794750663290
0.02188	0.9999999809194794441132744	0.9999999809194794429551550
0.02500	0.9999999674499511188938822	0.9999999674499511169717123
0.02813	0.9999999478619811395538211	0.9999999478619811365239619
0.03125	0.9999999205349010051448672	0.9999999205349010005561997

In this table we present a side-by-side evaluation of the exact solutions and the numerical solutions obtained using the proposed method across various time points t.

At t=0.003125 the exact solution is approximately 0.99999999999205272179096580, while the numerical solution yields 0.9999999999920527217876255. The difference between the two solutions is minimal, indicating high accuracy in the numerical method.

As we progress to t=0.00625, the exact solution is 0.9999999998728439211837312, with the numerical solution being 0.99999999987284392115939050. Again, the numerical method closely approximates the exact value, reflecting the reliability of the new approach.

At t=0.00938, the exact value is 0.9999999993562754941897026 compared to the numerical solution of 0.9999999993562754941109965. This demonstrates that the new method maintains its precision even as the time increment increases.

For t=0.01250 the exact solution is 0.9999999979655265806035874 with the numerical result at 0.9999999979655265804224698. The closeness of these values reaffirms the effectiveness of the numerical technique.

At t=0.01563 the exact solution is 0.999999995033067533444110 3while the numerical result is 0.9999999950330675330875725. The small difference here continues to illustrate the method's strong performance.

For t=0.01875, the exact value stands at 0.9999999897006794757253819 against the numerical result of 0.9999999897006794750663290. The comparison shows that the numerical solution remains very close to the exact solution.

At t = 0.02188, the exact solution is 0.9999999809194794441132744, with the numerical solution coming in at 0.9999999809194794429551550. This again highlights the method's accuracy.

At t = 0.02500, the exact solution reads 0.9999999674499511188938822 and the numerical result is 0.9999999674499511169717123, which indicates a continued high level of precision.

For t=0.02813. the exact value is 0.9999999478619811395538211, while the numerical solution is 0.9999999478619811365239619. The two solutions are still remarkably close, showcasing the consistency of the numerical method.

Finally, at t=0.03125, the exact solution is 0.9999999205349010051448672, and the numerical result is 0.9999999205349010005561997. The persistent closeness of the numerical values to the exact solutions throughout the table reinforces the robustness and effectiveness of the new numerical method applied to the problem.

Overall, the numerical solutions demonstrate exceptional accuracy when compared to the exact solutions, with negligible discrepancies at each time point, validating the reliability of the numerical approach in solving the problem at hand.

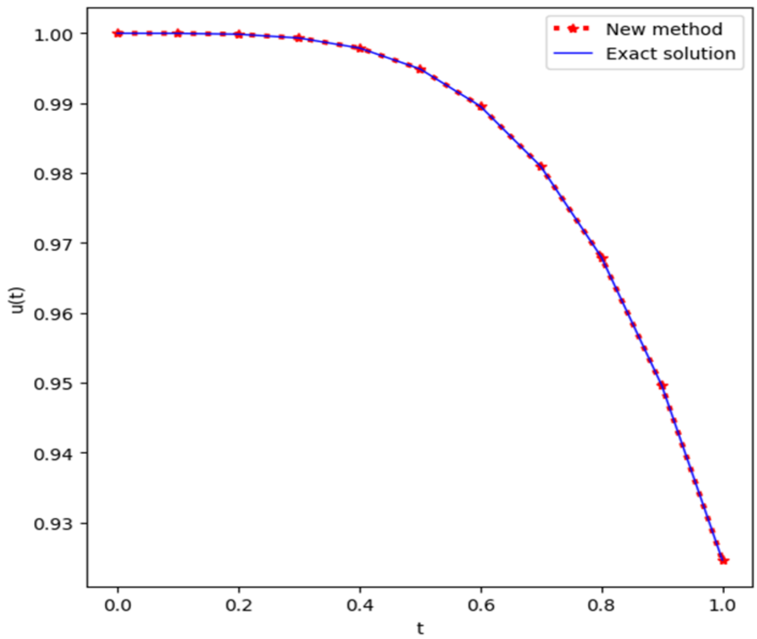


Figure 2. Comparison of exact and numerical solution of problem 3.2

The x-axis represents the time variable t, which starts at 0.003125 and increases in uniform steps up to 0.03125. The y-axis shows the values of the exact and numerical solutions, both of which are close to 1. The exact solution gradually decreases from 0.999999999992 at t=0.003125 to 0.999999920535 at t=0.03125. This indicates a very slow decline in the exact solution values over time. The numerical solution derived using the proposed method, also decreases in the same trend. At t = 0.003125, the numerical value is 0.999999999992, closely matching the exact solution. As t increases to 0.03125, the numerical solution reaches 0.999999920535, once again closely matching the exact solution. The difference between the exact and numerical solutions is minimal. For instance, at t = 0.003125 the exact solution is 0.99999999999920527217909658, and the numerical solution is 0.99999999999920527217876255, resulting in an insignificant difference of about 3.34×10^{-12}. This minimal error persists throughout the time interval, demonstrating the high accuracy of the proposed method. Both the exact and numerical solutions appeared as two nearly indistinguishable curves descending slowly over time. The lines almost overlapped due to the very small differences between the two solutions. The proposed method effectively tracks the exact solution with high precision. The graph showcases that for small time intervals, the numerical solution remains an excellent approximation of the exact solution, with negligible error.

Table 5. Comparison of Accuracy for Problem 3.2

T	Exact Solution	Error in [14]	Error in BHBDF IV
0.003125	0.99999167499652860438	1.900×10^{-19}	9.662×10^{-22}
0.00625	0.99986719911195714198	2.300×10^{-19}	3.731×10^{-21}
0.00938	0.99933105226749824584	8.600×10^{-19}	2.973×10^{-22}
0.01250	0.99790057330915505191	1.380×10^{-18}	3.587×10^{-21}
0.01563	0.99492052670511528095	3.53×10^{-18}	4.110×10^{-21}
0.01875	0.98958301770794685383	5.310×10^{-18}	4.618×10^{-21}
0.02188	0.98095229007588226219	8.88×10^{-18}	3.274×10^{-21}
0.02500	0.96799382462962246363	3.922×10^{-17}	3.882×10^{-21}
0.02813	0.94960705358355858549	5.846×10^{-17}	3.822×10^{-21}
0.03125	0.92466091697090496134	8.477×10^{-17}	4.867×10^{-21}

Table 5 presents data comparing the exact solution and the errors in two numerical methods: [14] and the Block Hybrid Backward Differentiation Formula IV (BHBDF IV). The first column (T) represents time, which increases incrementally from 0.003125 to 0.03125. The second column gives the exact solution at each corresponding time point. The third column shows the error values associated with the method [14]. The fourth column presents the error values for the BHBDF IV method. The exact solution gradually decreases as time progresses. For example, at T = 0.003125, the exact solution is approximately 0.99999 and by T = 0.03125, it has decreased to approximately 0.92466. This indicates that the function being modeled shows a steady decline over time. The error values for the method [14] start small at T = 0.00312 approximately 1.9×10^{-19} and gradually increase with time.

By T = 0.03125, the error has grown significantly to 8.477×10^{-17}. This shows that the accuracy of the Ukpebor method decreases as time progresses, with the error becoming more pronounced. The error values for the BHBDF

IV method start even smaller than Ukpebor's method at T = 0.003125 approximately 9.662×10^{-22}. Throughout the time interval, the BHBDF IV method maintains very low error values. For example, at T = 0.03125 the error is only 4.867×10^{-21} which is significantly smaller than the corresponding error in Ukpebor's method.

For [14], the error starts small but increases rapidly as time increases, indicating that the accuracy of this method diminishes over time. For the BHBDF IV method it consistently provides extremely small error values over the entire time range, showcasing a much higher accuracy compared to Ukpebor's method. The largest error for BHBDF IV at T = 0.03125 is still on the order of 10^{-21}which is substantially lower than Ukpebor's largest error of 10^{-17}. The table clearly demonstrates the superior accuracy of the BHBDF IV method. While the error in [14] grows significantly as time progresses, the error in the BHBDF IV method remains very small, making it a more reliable and accurate method for solving the given problem.

4. Conclusion

In this paperwork, the aim is to seek an effective numerical approach for solving higher-order ordinary differential equations. To accomplish this objective, the research focuses on developing a class of block hybrid backward differentiation formulas (BHBDFs) that can generate simultaneous approximate solutions for equations across the entire integration interval. The method is derived using the techniques of interpolation and collocation. During the derivation process of the method, specifically a four-step process (k = 4), a set of off-grid points were meticulously selected at the interpolation points across the interval [0, 4] for fourth-order ordinary differential (ODEs) BHBDF IV. Convergence analysis of the methods reveals that BHBDF IV is of order 9. The methods are zero stable and consistent which implies its convergence. Furthermore, numerical experiments were carried out where BHBDF IV was implemented on two problems. Comparative analyses on BHBDF IV in tables 1- 4 reveal that the method has an advantage of producing smaller global errors over several existing methods in the literature including some most recent ones. It also shows the efficacy of the technique.

Declaration of Competing Interest

No conflict of interest was declared by the authors.

Authorship Contribution Statement

Raihanatu Muhammad: Design the research work, develop the new method with interpretation, critical review.
Hajara Hussaini: Literature review, implementation the work, critical review
Abdulmalik Oyedeji: Writing, interpretation of the results, critical review

References

[1] W. Clarence de Silva, *Modeling of Dynamic Systems with Engineering Applications*, 2nd ed. CRC Press, 2022.
[2] W.E. Boyce, R.C. DiPrima, and D.B. Meade, *Elementary Differential Equations and Boundary Value Problems*, John Wiley & Sons, 2021.
[3] E.W. Schiesser and G.W. Griffiths, *A Compendium of Partial Differential Equations: Model Method of Lines Analysis with MATLAB*, Cambridge University Press, 2009.
[4] G.D. Bryne and J.D. Lambert, "Pseudorunge-Kutta methods involving two points," *J. Assoc. Comput. Mathematics*, vol. 40, pp. 114–123, Jan. 1996, doi: 10.1145/321312.321321.
[5] A. J. Salgado and S. M. Wise, "Classical Numerical Analysis: A Comprehensive Course," *Cambridge University Press*, 2022, ch. 20, p. 555-556.
[6] G. Söderlind, I. Fekete, and I. Faragó, "On the zero-stability of multistep methods on smooth nonuniform grids," *SIAM Journal on Numerical Analysis*, vol. 58, pp. 1125–1143, Dec. 2018, doi: 10.1007/s10543-018-0716-y

[7] R. D’Ambrosio, "Linear Multistep Methods," *Numerical Approximation of Ordinary Differential Problems*, vol. 148, Springer, Cham, 2023.

[8] A.C. Cardone and R. D'Ambrosio, “Collocation methods for Volterra integral and integro-differential equations: A review,” Department of Mathematics, University of Salerno, 84084 Fisciano, 2018.

[9] A. B. Familua and E. O. Omole, “Five Points Mono Hybrid Point Linear Multistep Method for Solving Nth Order Ordinary Differential Equations Using Power Series Function,” Asian Res. J. Math., vol. 3, no. 1, pp. 1–17, Jan. 2017, doi: 10.9734/ARJOM/2017/31190.

[10] E.O. Adeyefa and J.O. Kuboye, “Derivation of new numerical model capable of solving second and third order ordinary differential equations directly,” *IEANG Int. J. of Applied Mat.*, vol. 50, no. 2, pp. 1-9, Jun. 2020.

[11] J.O. Kuboye, O.R. Elusakin, and O.F. Quadri, “Numerical algorithm for direct solution of fourth order ordinary differential equations,” *J. Nigerian Society of Physical Sciences*, vol. 2, pp. 218–227, Nov. 2020, doi: 10.46481/jnsps.2020.100.

[12] K. J. Audu, J. Garba, A. T. Tiamiyu, and B. A. Thomas, "Application of Backward Differentiation Formula on Fourth-Order Differential Equations," *J. Sci.Techno.*, vol. 14, no. 2, pp. 52–65, Dec. 2022, doi: 10.30880/jst.2022.14.02.006.

[13] O.E. Abolarin and B.G. Ogunware, “New hybrid method for direct numerical solution of nonlinear second, third and fourth orders ordinary differential equations,” *Int. J. Mathematics in Operational Research*, vol. 23, no. 3, pp. 285-315, Nov. 2022, doi.org/10.1504/IJMOR.2022.127378

[14] L. Ukpebor and O. Ezekiel, “Three-step optimized backward differentiation formulae (TOBBDF) for solving stiff ordinary differential equations,” *African Journal of Mathematics and Computer Science Research*, Vol. 13, No.1, pp. 51-57, Apr. 2020. doi: 10.5897/AJMCSR2019.0811.