Journal of Nonlinear Dynamics and Applications
http://dx.doi.org/10.62762/JNDA.2026.749084

RESEARCH ARTICLE

A Third-Order Variable Step Size Superclass of Block Backward Differentiation Formula for Efficient Solution of Highly Stiff Differential Systems

Najamuddeen Bala[1,*], **Hamisu Musa**[2] **and Buhari Alhassan**[3]

[1] Department of Statistics, Federal Polytechnic Kaura-Namoda, Zamfara State, Nigeria
[2] Department of Mathematics, Umaru Musa Yar'adua University, Katsina, Katsina State, Nigeria
[3] Department of Mathematics and Statistics, Al-Qalam University Katsina, Katsina State, Nigeria

Abstract

This paper develops a third-order fully implicit adaptive variable step-size superclass block backward differentiation formula (VSBBDF3) for the efficient numerical simulation of nonlinear stiff dynamical systems, including oscillatory, chaotic, and reaction-kinetics systems governed by ordinary differential equations. The method extends the classical block BDF framework through a structured superclass coefficient formulation while preserving full implicitness. By computing multiple solution approximations simultaneously within each block, the scheme enhances both stability and computational efficiency. An adaptive step-size strategy controls local truncation errors, enabling dynamic response to rapidly varying stiff behavior. Rigorous theoretical analysis establishes consistency, zero-stability, convergence, and A-stability, confirming their reliability for stiff problems. Nonlinear systems from the implicit formulation are efficiently handled using Newton-type iteration. Extensive numerical experiments on stiff linear, oscillatory, and nonlinear problems demonstrate that the proposed method consistently achieves higher accuracy with competitive computational cost compared to existing methods, including NBDF, VSBBDF, and MATLAB ODE solvers. The results further show that combining block formulation, superclass structure, and adaptive step-size control provides a more effective accuracy-efficiency balance. Overall, the VSBBDF3 method offers a robust, accurate, and efficient framework for the numerical simulation of nonlinear stiff dynamical systems, well suited for scientific and engineering applications in dynamics, control, and bifurcation analysis.

Keywords: nonlinear dynamical systems, variable step size, superclass, block backward differentiation formula, convergence.

1 Introduction

The numerical treatment of ordinary differential equations (ODEs) governing nonlinear dynamical systems is a central problem in scientific computation, with wide-ranging applications in fields such as physics, engineering, chemistry, biology, and economics. Although many real-world models

Submitted: 24 February 2026
Revised: 18 May 2026
Accepted: 24 June 2026
Published: 10 August 2026

Vol. 2, **No.** 3, 2026.
10.62762/JNDA.2026.749084

***Corresponding author:**
Najamuddeen Bala
phdmth190179@students.umyu.edu.ng

Citation
Bala, N., Musa, H., & Alhassan, B. (2026). A Third-Order Variable Step Size Superclass of Block Backward Differentiation Formula for Efficient Solution of Highly Stiff Differential Systems. *Journal of Nonlinear Dynamics and Applications*, 2(3), 143–158.

can be described using nonstiff systems—for which classical explicit solvers perform well [1]—a significant number of practical phenomena are stiff, meaning they involve processes operating on widely different time scales. Such stiffness commonly occurs in areas including chemical reaction kinetics, electrical circuit simulation, climate dynamics, and control engineering. A key challenge posed by stiff systems is that explicit numerical schemes often demand extremely small step sizes to preserve stability, which in turn results in high computational expense and reduced efficiency [2, 3]. Nonlinear dynamical systems governed by differential equations frequently arise in many scientific and engineering disciplines, where their numerical trajectories must be accurately reproduced over extended time intervals. However, nonlinear stiffness often introduces multiple time scales and creates significant challenges for conventional integration techniques. Therefore, developing stable and efficient numerical schemes for nonlinear stiff dynamical systems remains an important problem in computational dynamics.

Implicit numerical schemes, especially backward differentiation formulas (BDFs), are well known for their effectiveness in solving stiff differential equations because of their strong stability characteristics. Standard BDF methods are multistep techniques that estimate derivatives through backward finite differences, producing stability regions that cover substantial parts of the left half of the complex plane. Despite these advantages, conventional BDF approaches advance the solution step-by-step in time, which reduces their suitability for parallel computation and can restrict efficiency when dealing with large-scale problems or when highly accurate solutions are needed [4–6].

To address these limitations, block methods have been proposed, in which the solution is approximated simultaneously at several points within an integration block. Block methods can leverage modern computational architectures and reduce overall function evaluations by exploiting intra-block structure. Among these, block backward differentiation formulas (BBDFs), block extended backward differentiation formulas (BEBDFs), and superclass of block extended backward differentiation formulas (SBEBDFs) compute a block of solution values in a single update, thereby improving efficiency and providing a framework that naturally accommodates implicitness and stability requirements [7, 8]. Despite the advantages of block formulations, there remains a need for systematic generalization and refinement of these methods to achieve higher accuracy and better stability for both stiff and oscillatory problems.

In the context of block methods, the superclass approach introduces a parameterized framework that generalizes a family of block formulas. Block formulations such as continuous block BDF have demonstrated the flexibility of coefficient parameterization for improved accuracy [9], while foundational stability theory for linear multistep methods—particularly the barrier on A-stable methods—constrains the design space of any such scheme [10]. Superclass block formulas exploit this design space by allowing control over method coefficients to produce schemes with enhanced stability properties and reduced truncation error constants. Although several superclass block methods have been proposed in the literature, most existing formulations are restricted to fixed time steps or do not fully exploit the potential of variable step-size strategies [11].

Variable step-size methods are crucial for efficiently handling problems with varying solution behaviors, such as transient layers followed by slow dynamics. A robust variable step-size algorithm adapts the step length based on local error estimates or stability requirements, ensuring both accuracy and computational efficiency. However, combining variable step-size control with block and superclass formulations poses significant theoretical and practical challenges, including preservation of stability properties such as A-stability and zero-stability under step-size variation [12–14].

Although several numerical methods exist for solving stiff ODEs, many classical schemes such as BDF methods are sequential and computationally inefficient, while existing block and superclass methods are often limited to fixed step sizes or do not fully integrate adaptive control with high-order implicit formulations. The gap in the literature is the absence of a third order fully implicit, variable step-size superclass block method that simultaneously ensures high accuracy, strong stability, and computational efficiency within a unified framework. To address this, this study proposes a novel Third Order Fully Implicit Variable Step-Size Superclass Block Backward Differentiation Formula (VSBBDF3), which uniquely combines superclass parameterization, block implicit formulation, and adaptive step-size control in a single scheme for stiff ODEs.

The primary objective of this study is to formulate and analyze VSBBDF3, a new implicit block integration scheme that integrates the strengths of superclass formulations, block structure, and variable step-size control. The proposed method is designed to deliver high accuracy (third-order global convergence), strong stability characteristics suitable for stiff ODEs, and computational efficiency.

2 Formulation of the Method

Definition 1. The third order fully implicit variable step size super class of block backward differentiation formula (VSBBDF3) method is defined as:

$$\sum_{k=0}^{3} \sigma_{k,i,r} v_{n+k-1} = h\Psi_{j,i,r}(f_{n+j} - \rho f_{n+j-1}),\ j = i = 1,2. \tag{1}$$

Definition 2. The Linear operator α_i associated with the VSBBDF3 is defined by:

$$\begin{aligned}\alpha_i[v(u_n),h] &:= \sigma_{0,i}v_{n-1} + \sigma_{1,i}v_n + \sigma_{2,i}v_{n+1} + \sigma_{3,i}v_{n+2} \\ &- h\Psi_{k,i,r}(v'_{n+j} - \rho v'_{n+j-1}) = 0, \quad j = i = 1,2.\end{aligned} \tag{2}$$

The derivation of the method is presented in two cases. Case 1 corresponds to the derivation of the first point, for which $j = i = 1$, while Case 2 addresses the derivation of the second point.

2.1 Derivation of the First Point (Case 1): $j = i = 1$

To derive the first point v_{n+1}, we define the associated linear operator as:

$$\begin{aligned}\alpha_1[v(u_n),h] :&= \sigma_{0,1}v(u_n - rh) + \sigma_{1,1}v(u_n) \\ &+ \sigma_{2,1}v(u_n + h) + \sigma_{3,1}v(u_n + 2h) \\ &- h\Psi_{1,1}(v'(u_n + h) - \rho v'(u_n)) = 0,\end{aligned} \tag{3}$$

Expanding equation (3) in a Taylor series around u_n and grouping terms of the same order results in the following expression:

$$\begin{aligned}&\phi_{0,1}v(u_n) + \phi_{1,1}hv'(u_n) + \phi_{2,1}h^2v''(u_n) \\ &+ \phi_{3,1}h^3v'''(u_n) + \cdots = 0,\end{aligned} \tag{4}$$

where,

$$\begin{aligned}\phi_{0,1} &= \sigma_{0,1} + \sigma_{1,1} + \sigma_{2,1} + \sigma_{3,1} = 0 \\ \phi_{1,1} &= -r\sigma_{0,1} + \sigma_{2,1} + 2\sigma_{3,1} - \Psi_{1,1}(1-\rho) = 0 \\ \phi_{2,1} &= \frac{1}{2}r^2\sigma_{0,1} + \frac{1}{2}\sigma_{2,1} + 2\sigma_{3,1} - \Psi_{1,1} = 0 \\ \phi_{3,1} &= -\frac{1}{6}r^3\sigma_{0,1} + \frac{1}{6}\sigma_{2,1} + \frac{4}{3}\sigma_{3,1} - \frac{1}{2}\Psi_{1,1} = 0\end{aligned} \tag{5}$$

In the derivation of the first point v_{n+1}, the coefficient $\sigma_{2,1}$ is normalized to unity. Solving the resulting system of simultaneous equations in (5) using Maple 18 yields the coefficients $\sigma_{j,i}$ and $\Psi_{j,i}$ as follows. The coefficients for the first point are summarized in Table 1. Substituting these values in equation (3), we obtain

Table 1. Coefficient of the first point.

Coefficient	Expression
$\sigma_{0,1}$	$-\dfrac{2\rho+1}{2r^2\rho+4r\rho-r-2}$
$\sigma_{1,1}$	$-\frac{1}{2}\dfrac{3r^2\rho-r^2+r\rho-2r-2\rho-1}{r(2r\rho-1)}$
$\sigma_{2,1}$	1
$\sigma_{3,1}$	$-\frac{1}{2}\dfrac{r^2\rho+r^2+r\rho+2r+1}{2r^2\rho+4r\rho-r-2}$
$\Psi_{1,1}$	$-\dfrac{r+1}{2r\rho-1}$

$$\begin{aligned}v_{n+1} = &\frac{2\rho+1}{(2r^2\rho+4r\rho-r-2)}v_{n-1} \\ &+ \frac{1}{2}\frac{3r^2\rho-r^2+r\rho-2r-2\rho-1}{r(2r\rho-1)}v_n \\ &+ \frac{1}{2}\frac{r^2\rho+r^2+r\rho+2r+1}{2r^2\rho+4r\rho-r-2}v_{n+2} \\ &- \frac{r+1}{2r\rho-1}hf_{n+1} + \frac{r+1}{2r\rho-1}hf_n.\end{aligned} \tag{6}$$

2.2 Derivation of the Second Point (Case 2): $j = i = 2$

To obtain the second point v_{n+2}, the corresponding linear operator is defined as follows:

$$\begin{aligned}\alpha_2[v(u_n),h] = &\sigma_{0,2}v(u_n - rh) + \sigma_{1,2}v(u_n) + \sigma_{2,2}v(u_n + h) \\ &+ \sigma_{3,2}v(u_n + 2h) - h\Psi_{2,2}(v'(u_n + 2h) \\ &- \rho v'(u_n + h)) = 0,\end{aligned} \tag{7}$$

Expanding equation (7) as a Taylor series about u_n and collecting the like terms gives

$$\begin{aligned}&\phi_{0,2}v(u_n) + \phi_{1,2}hv'(u_n) + \phi_{2,2}h^2v''(u_n) \\ &+ \phi_{3,2}h^3v'''(u_n) + \cdots = 0,\end{aligned} \tag{8}$$

where,

$$\begin{aligned}
\phi_{0,2} &= \sigma_{0,2} + \sigma_{1,2} + \sigma_{2,2} + \sigma_{3,2} = 0 \\
\phi_{1,2} &= -r\sigma_{0,2} + \sigma_{2,2} + 2\sigma_{3,2} - \Psi_{2,2}(1-\rho) = 0 \\
\phi_{2,2} &= \frac{1}{2}r^2\sigma_{0,2} + \frac{1}{2}\sigma_{2,2} + 2\sigma_{3,2} - \Psi_{2,2}(2-\rho) = 0 \\
\phi_{3,2} &= -\frac{1}{6}r^3\sigma_{0,2} + \frac{1}{6}\sigma_{2,2} + \frac{4}{3}\sigma_{3,2} - \frac{1}{2}\Psi_{2,2}\left(2 - \frac{1}{2}\rho\right) = 0
\end{aligned} \tag{9}$$

In the derivation of the second point v_{n+2}, the coefficient $\sigma_{3,2}$ is normalized to unity. Solving the system of simultaneous equations in (9) yields the coefficients $\sigma_{j,i}$ and $\Psi_{j,i}$ as follows. The coefficients for the second point are given in Table 2.

Table 2. Coefficient of the second point.

Coefficient	Expression
$\sigma_{0,2}$	$\dfrac{2(\rho+2)}{r(r\rho-3r+\rho-8)(r+1)}$
$\sigma_{1,2}$	$\dfrac{r^2\rho+r^2+3r\rho+4r+2\rho+4}{-r(r\rho-3r+\rho-8)(r+1)}$
$\sigma_{2,2}$	$\dfrac{2(2r^2+r\rho+8r+2\rho+8)}{(r\rho-3r+\rho-8)(r+1)}$
$\sigma_{3,2}$	1
$\Psi_{2,2}$	$-\dfrac{2(r+2)}{r\rho-3r+\rho-8}$

By inserting these values into equation (7), the resulting formulation is obtained:

$$\begin{aligned}
v_{n+2} = &-\frac{2(\rho+2)}{r(r\rho-3r+\rho-8)(r+1)}v_{n-1} \\
&+\frac{r^2\rho+r^2+3r\rho+4r+2\rho+4}{r(r\rho-3r+\rho-8)}v_n \\
&-\frac{2(2r^2+r\rho+8r+2\rho+8)}{(r\rho-3r+\rho-8)(r+1)}v_{n+1} \\
&-\frac{2(r+2)}{r\rho-3r+\rho-8}hf_{n+2} \\
&+\frac{2(r+2)}{r\rho-3r+\rho-8}\rho hf_{n+1}.
\end{aligned} \tag{10}$$

Hence, combining equations (6) and (10) leads to the derivation of the VSBBDF3 scheme, which is given as follows:

$$\begin{cases}
v_{n+1} = \dfrac{2\rho+1}{(2r^2\rho+4r\rho-r-2)}v_{n-1} \\
\quad + \dfrac{1}{2}\dfrac{3r^2\rho-r^2+r\rho-2r-2\rho-1}{r(2r\rho-1)}v_n \\
\quad + \dfrac{1}{2}\dfrac{r^2\rho+r^2+r\rho+2r+1}{2r^2\rho+4r\rho-r-2}v_{n+2} \\
\quad - \dfrac{r+1}{2r\rho-1}hf_{n+1} \\
\quad + \dfrac{r+1}{2r\rho-1}\rho hf_n, \\
v_{n+2} = -\dfrac{2(\rho+2)}{r(r\rho-3r+\rho-8)(r+1)}v_{n-1} \\
\quad + \dfrac{r^2\rho+r^2+3r\rho+4r+2\rho+4}{r(r\rho-3r+\rho-8)}v_n \\
\quad - \dfrac{2(2r^2+r\rho+8r+2\rho+8)}{(r\rho-3r+\rho-8)(r+1)}v_{n+1} \\
\quad - \dfrac{2(r+2)}{r\rho-3r+\rho-8}hf_{n+2} \\
\quad + \dfrac{2(r+2)}{r\rho-3r+\rho-8}\rho hf_{n+1}.
\end{cases} \tag{11}$$

The selection of the superclass parameter ρ is essential to the stability and efficiency of the proposed method. In this study, the value $\rho = -\frac{3}{4}$ is adopted based on established results in the literature, where it has been shown to produce stable and accurate numerical solutions for stiff differential systems [15–17].

This choice lies within the admissible interval $(-1, 1)$, ensuring that the method retains important properties such as zero-stability and A-stability. As a result, it provides a suitable balance between stability and accuracy while remaining consistent with existing methods.

With $\rho = -\frac{3}{4}$ fixed, varying the step-size ratio r in equation (11) leads to different forms of the third-order fully implicit variable step-size superclass of block backward differentiation formulas. The corresponding schemes for $r = 1$, $r = 2$ and $r = \frac{5}{8}$ are presented in equations (12)-(14), respectively:

For $r = 1$:

$$\begin{aligned}
v_{n+1} &= \frac{1}{15}v_{n-1} + \frac{11}{10}v_n - \frac{1}{6}v_{n+2} + \frac{4}{5}hf_{n+1} + \frac{3}{5}hf_n \\
v_{n+2} &= \frac{1}{10}v_{n-1} - \frac{9}{25}v_n + \frac{63}{50}v_{n+1} + \frac{12}{25}hf_{n+2} + \frac{9}{25}hf_{n+1}
\end{aligned} \tag{12}$$

For $r = 2$:

$$\begin{aligned}
v_{n+1} &= \frac{1}{64}v_{n-1} + \frac{9}{8}v_n - \frac{9}{64}v_{n+2} + \frac{3}{4}hf_{n+1} + \frac{9}{16}hf_n \\
v_{n+2} &= \frac{1}{39}v_{n-1} - \frac{14}{65}v_n + \frac{232}{195}v_{n+1} + \frac{32}{65}hf_{n+2} + \frac{24}{65}hf_{n+1}
\end{aligned} \tag{13}$$

For $r = \frac{5}{8}$:

$$v_{n+1} = \frac{512}{3255}v_{n-1} + \frac{637}{620}v_n - \frac{481}{2604}v_{n+2} + \frac{26}{31}hf_{n+1} + \frac{39}{62}hf_n$$
$$v_{n+2} = \frac{1024}{4615}v_{n-1} - \frac{189}{355}v_n + \frac{6048}{4615}v_{n+1} + \frac{168}{355}hf_{n+2} + \frac{126}{355}hf_{n+1} \tag{14}$$

The implementation of the method adopts a predictor-corrector framework. In particular, the corrector is executed in PECE mode, whereby the next solution value is predicted, evaluated, corrected, and subsequently re-evaluated to improve the overall accuracy of the method.

3 Order of the Method

In this section, we determine the order of the methods given by (12), (13), and (14). To begin, we focus on the formula in (12), which can alternatively be expressed in the following form:

For $r = 1$:

$$-\frac{1}{15}v_{n-1} - \frac{11}{10}v_n + v_{n+1} + \frac{1}{6}v_{n+2} = \frac{4}{5}hf_{n+1} + \frac{3}{5}hf_n$$
$$-\frac{1}{10}v_{n-1} + \frac{9}{25}v_n - \frac{63}{50}v_{n+1} + v_{n+2} = \frac{12}{25}hf_{n+2} + \frac{9}{25}hf_{n+1} \tag{15}$$

The corresponding matrix representation of equation (15) can be written as follows:

$$\begin{bmatrix} -\frac{1}{15} & -\frac{11}{10} \\ -\frac{1}{10} & \frac{9}{25} \end{bmatrix} \begin{bmatrix} v_{n-1} \\ v_n \end{bmatrix} + \begin{bmatrix} 1 & \frac{1}{6} \\ -\frac{63}{50} & 1 \end{bmatrix} \begin{bmatrix} v_{n+1} \\ v_{n+2} \end{bmatrix} = h\begin{bmatrix} 0 & \frac{3}{5} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix} + h\begin{bmatrix} \frac{4}{5} & 0 \\ \frac{9}{25} & \frac{12}{25} \end{bmatrix} \begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix} \tag{16}$$

The general matrix representation of equation (16) is given by:

$$\sum_{k=0}^{1} \gamma_k^* V_{m+k-1} = h\sum_{k=0}^{1} \Gamma_k^* F_{m+k-1} \tag{17}$$

where $\gamma_0^*, \gamma_1^*, \Gamma_0^*, \Gamma_1^*$ are 2×2 square matrices defined by:

$$\gamma_0^* = \begin{bmatrix} -\frac{1}{15} & -\frac{11}{10} \\ -\frac{1}{10} & \frac{9}{25} \end{bmatrix}, \quad \gamma_1^* = \begin{bmatrix} 1 & \frac{1}{6} \\ -\frac{63}{50} & 1 \end{bmatrix},$$
$$\Gamma_0^* = \begin{bmatrix} 0 & \frac{3}{5} \\ 0 & 0 \end{bmatrix}, \quad \Gamma_1^* = \begin{bmatrix} \frac{4}{5} & 0 \\ \frac{9}{25} & \frac{12}{25} \end{bmatrix}.$$

and V_{m-1}, V_m, F_{m-1} and F_m are column vectors defined by:

$$V_{m-1} = \begin{bmatrix} v_{n-1} \\ v_n \end{bmatrix}, \quad V_m = \begin{bmatrix} v_{n+1} \\ v_{n+2} \end{bmatrix},$$
$$F_{m-1} = \begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix}, \quad F_m = \begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix}.$$

Let $\gamma_0^*, \gamma_1^*, \Gamma_0^*, \Gamma_1^*$ be block matrices defined by:

$$\gamma_0^* = \begin{bmatrix} \gamma_0 & \gamma_1 \end{bmatrix}, \quad \gamma_1^* = \begin{bmatrix} \gamma_2 & \gamma_3 \end{bmatrix},$$
$$\Gamma_0^* = \begin{bmatrix} \Gamma_0 & \Gamma_1 \end{bmatrix}, \quad \Gamma_1^* = \begin{bmatrix} \Gamma_2 & \Gamma_3 \end{bmatrix}.$$

where,

$$\gamma_0 = \begin{bmatrix} -\frac{1}{15} \\ -\frac{1}{10} \end{bmatrix}, \gamma_1 = \begin{bmatrix} -\frac{11}{10} \\ \frac{9}{25} \end{bmatrix}, \gamma_2 = \begin{bmatrix} 1 \\ -\frac{63}{50} \end{bmatrix}, \gamma_3 = \begin{bmatrix} \frac{1}{6} \\ 1 \end{bmatrix},$$
$$\Gamma_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \Gamma_1 = \begin{bmatrix} \frac{3}{5} \\ 0 \end{bmatrix}, \quad \Gamma_2 = \begin{bmatrix} \frac{4}{5} \\ \frac{9}{25} \end{bmatrix}, \quad \Gamma_3 = \begin{bmatrix} 0 \\ \frac{12}{25} \end{bmatrix}.$$

Definition 3. The order of the block method (12) and its associated linear difference operator given by

$$\alpha[v(u); h] = \sum_{k=0}^{3} [\gamma_k v(u + kh) - h\Gamma_k v'(u + kh)]. \tag{18}$$

is the unique integer p such that the difference operator (18) and the associated method (12) is considered of order p if $\delta_0 = \delta_1 = \delta_2 = \ldots = \delta_p = 0$ and $\delta_{p+1} \neq 0$ [18–20].

where,

$$\delta_0 = \sum_{k=0}^{3} \gamma_k = \gamma_0 + \gamma_1 + \gamma_2 + \gamma_3$$
$$= \begin{bmatrix} -\dfrac{1}{15} \\ -\dfrac{1}{10} \end{bmatrix} + \begin{bmatrix} -\dfrac{11}{10} \\ \dfrac{9}{25} \end{bmatrix} + \begin{bmatrix} 1 \\ -\dfrac{63}{50} \end{bmatrix} + \begin{bmatrix} \dfrac{1}{6} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

$$\delta_1 = \sum_{k=0}^{3} (k\gamma_k - \Gamma_k)$$
$$= \big((0)\gamma_0 + (1)\gamma_1 + (2)\gamma_2 + (3)\gamma_3\big) - (\Gamma_0 + \Gamma_1 + \Gamma_2 + \Gamma_3)$$
$$= \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

$$\delta_2 = \sum_{k=0}^{3} \left(\frac{1}{2!} k^2 \gamma_k - \frac{1}{1!} k \Gamma_k \right)$$
$$= \frac{1}{2!}\big((0)^2\gamma_0 + (1)^2\gamma_1 + (2)^2\gamma_2 + (3)^2\gamma_3\big) - \frac{1}{1!}\big((0)\Gamma_0 + (1)\Gamma_1 + (2)\Gamma_2 + (3)\Gamma_3\big)$$
$$= \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

$$\delta_3 = \sum_{k=0}^{3} \left(\frac{1}{3!} k^3 \gamma_k - \frac{1}{2!} k^2 \Gamma_k \right)$$
$$= \frac{1}{3!}\big((0)^3\gamma_0 + (1)^3\gamma_1 + (2)^3\gamma_2 + (3)^3\gamma_3\big) - \frac{1}{2!}\big((0)^2\Gamma_0 + (1)^2\Gamma_1 + (2)^2\Gamma_2 + (3)^2\Gamma_3\big)$$
$$= \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

$$\delta_4 = \sum_{k=0}^{3} \left(\frac{1}{4!} k^4 \gamma_k - \frac{1}{3!} k^3 \Gamma_k \right)$$
$$= \frac{1}{4!}\big((0)^4\gamma_0 + (1)^4\gamma_1 + (2)^4\gamma_2 + (3)^4\gamma_3\big) - \frac{1}{3!}\big((0)^3\Gamma_0 + (1)^3\Gamma_1 + (2)^3\Gamma_2 + (3)^3\Gamma_3\big)$$
$$= \begin{bmatrix} \frac{1}{60} \\ -\frac{3}{100} \end{bmatrix}.$$

Therefore, the method (12) is of order 3, with error constant

$$\delta_4 = \begin{bmatrix} \frac{1}{60} \\ -\frac{3}{100} \end{bmatrix} \neq \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Applying the same procedure to methods (13) and (14), it can be shown that both are of order three, with their respective error constants given as follows:

$$\delta_4 = \begin{bmatrix} -\frac{9}{78} \\ -\frac{17}{153} \end{bmatrix} \neq \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \delta_4 = \begin{bmatrix} -\frac{76}{94} \\ -\frac{93}{267} \end{bmatrix} \neq \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

4 Stability Analysis of the Method

This section investigates the stability characteristics of methods (12)-(14), with particular emphasis on zero-stability and A-stability. The analysis is carried out by examining the roots of the associated characteristic (stability) polynomial and their positions relative to the unit circle in the complex plane. For each method, the stability region is derived using the standard linear test equation $v' = \lambda v$, where λ is a complex parameter satisfying $\Re(\lambda) < 0$. The procedure is first demonstrated for formula (12), after which the same approach is extended to formulas (13) and (14). Substituting the test equation $v' = \lambda v$, with $\Re(\lambda) < 0$, into formula (12) yields:

$$v_{n+1} = \frac{1}{15} v_{n-1} + \frac{11}{10} v_n - \frac{1}{6} v_{n+2} + \frac{4}{5} h\lambda v_{n+1} + \frac{3}{5} h\lambda v_n,$$
$$v_{n+2} = \frac{1}{10} v_{n-1} - \frac{9}{25} v_n + \frac{63}{50} v_{n+1} + \frac{12}{25} h\lambda v_{n+2} + \frac{9}{25} h\lambda v_{n+1}. \tag{19}$$

By rearranging the expression and grouping like terms in (19), we obtain:

$$v_{n+1} - \frac{4}{5} h\lambda v_{n+1} + \frac{1}{6} v_{n+2} = \frac{1}{15} v_{n-1} + \frac{11}{10} v_n + \frac{3}{5} h\lambda v_n$$
$$-\frac{63}{50} v_{n+1} - \frac{9}{25} h\lambda v_{n+1} + v_{n+2} - \frac{12}{25} h\lambda v_{n+2} = \frac{1}{10} v_{n-1} - \frac{9}{25} v_n \tag{20}$$

These equations can be expressed in matrix form as follows:

$$\begin{bmatrix} (1 - \frac{4}{5}\lambda h) & \frac{1}{6} \\ (-\frac{63}{50} - \frac{9}{25}\lambda h) & (1 - \frac{12}{25}\lambda h) \end{bmatrix} \begin{bmatrix} v_{n+1} \\ v_{n+2} \end{bmatrix} = \begin{bmatrix} \frac{1}{15} & (\frac{11}{10} + \frac{3}{5}\lambda h) \\ \frac{1}{10} & -\frac{9}{25} \end{bmatrix} \begin{bmatrix} v_{n-1} \\ v_n \end{bmatrix}. \tag{21}$$

Let $H = \lambda h$ in equation (21). This leads to

$$\begin{bmatrix} (1 - \frac{4}{5}H) & \frac{1}{6} \\ (-\frac{63}{50} - \frac{9}{25}H) & (1 - \frac{12}{25}H) \end{bmatrix} \begin{bmatrix} v_{n+1} \\ v_{n+2} \end{bmatrix} = \begin{bmatrix} \frac{1}{15} & (\frac{11}{10} + \frac{3}{5}H) \\ \frac{1}{10} & -\frac{9}{25} \end{bmatrix} \begin{bmatrix} v_{n-1} \\ v_n \end{bmatrix}. \tag{22}$$

Definition 4. If m is the number of block and y is the number of points in the block, then $n = my$. Here

$s = 2$ and $n = 2m$. By [21], we let

$$V_m = \begin{bmatrix} v_{n+1} \\ v_{n+2} \end{bmatrix} = \begin{bmatrix} v_{2m+1} \\ v_{2m+2} \end{bmatrix},$$
$$V_{m-1} = \begin{bmatrix} v_{n-1} \\ v_n \end{bmatrix} = \begin{bmatrix} v_{2m-1} \\ v_{2m} \end{bmatrix} = \begin{bmatrix} v_{2(m-1)-1} \\ v_{2(m-1)+2} \end{bmatrix}.$$

Equation (22) can be rewritten in the following form:

$$AV_m = BV_{m-1}. \tag{23}$$

where,

$$A = \begin{bmatrix} (1 - \frac{4}{5}H) & \frac{1}{6} \\ (-\frac{63}{50} - \frac{9}{25}H) & (1 - \frac{12}{25}H) \end{bmatrix},$$
$$B = \begin{bmatrix} \frac{1}{15} & (\frac{11}{10} + \frac{3}{5}H) \\ \frac{1}{10} & -\frac{9}{25} \end{bmatrix}. \tag{1}$$

The stability polynomial $\beta_1(\tau, H)$ of the method which is the combination of first and second characteristic polynomial is obtained by evaluating the determinant of $(A\tau - B)$ to obtain:

$$\begin{aligned} \beta(\tau, H) &= \det(A\tau - B) \\ &= \left| \begin{bmatrix} (1 - \frac{4}{5}H) & \frac{1}{6} \\ (-\frac{63}{50} - \frac{9}{25}H) & (1 - \frac{12}{25}H) \end{bmatrix} \tau - \begin{bmatrix} \frac{1}{15} & \frac{11}{10} + \frac{3}{5}H \\ \frac{1}{10} & -\frac{9}{25} \end{bmatrix} \right| \\ &= \frac{121}{100}\tau^2 - \frac{61}{50}\tau^2 H - \frac{269}{250}\tau + \frac{48}{125}\tau^2 H^2 \\ &\quad - \frac{176}{125}\tau H - \frac{67}{500} - \frac{27}{125}\tau H^2 - \frac{3}{50}H. \end{aligned} \tag{24}$$

The absolute stability region of the method (24) is determined by plotting the graph of

$$\beta_1(\tau, H) = 0, \tag{25}$$

$$\begin{aligned} \beta_1(\tau, H) = \frac{121}{100}\tau^2 - \frac{61}{50}\tau^2 H - \frac{269}{250}\tau + \frac{48}{125}\tau^2 H^2 \\ - \frac{176}{125}\tau H - \frac{67}{500} - \frac{27}{125}\tau H^2 - \frac{3}{50}H = 0. \end{aligned} \tag{26}$$

To establish the zero-stability of method (12), we set $H = 0$ in equation (26), which yields the first characteristic equation as follows:

$$\frac{121}{100}\tau^2 - \frac{269}{250}\tau - \frac{67}{500} = 0. \tag{27}$$

Solving equation (27) with respect to τ, the roots are obtained as follows:

$$\tau = 1, \quad \tau = -0.110744.$$

Definition 5 (Root Condition)**.** A first characteristics polynomial of a LMM is said to satisfy the root condition if all its roots lie within or on the unit circle, with those on the boundary being simple. In other words, the method is said to satisfy the root condition if all the roots τ_i of the polynomial satisfy $|\tau_i| \le 1$, $i = 1, 2, \dots, k$, and any root with modulus $|\tau_i| = 1$ is simple (i.e., has multiplicity one) [22].

Definition 6 (Zero-Stability)**.** A LMM is said to be zero-stable if its first characteristic polynomial satisfies the root condition. That is, the numerical solution remains bounded as the step size $h \to 0$, ensuring that small perturbations in the initial data do not lead to unbounded growth in the solution. In other words, a LMM is zero-stable if no root of the first characteristic polynomial lies outside the unit circle, and any root on the unit circle is simple (i.e., has multiplicity one) [4, 5].

Definition 7 (A-stability)**.** A LMM is said to be A-stable if the stability region covers the entire left half side of the complex plane [23].

Thus, in line with Definition 6, the computed values of τ show that method (12) is zero-stable, since all roots lie within or on the unit circle and the root $\tau = 1$ is simple. Adopting a similar procedure for the value of $r = 2$, the stability polynomial is obtained as:

$$\begin{aligned} \beta_2(\tau, H) = \frac{600}{520}\tau^2 - \frac{619}{520}\tau^2 H - \frac{2361}{2080}\tau + \frac{24}{65}\tau^2 H^2 \\ - \frac{161}{130}\tau H - \frac{67}{2080} - \frac{27}{130}\tau H^2 - \frac{3}{208}H = 0. \end{aligned} \tag{28}$$

For zero stability, we set $H = 0$ in equation (28) to obtain the first characteristics polynomial as

$$\frac{600}{520}\tau^2 - \frac{2361}{2080}\tau - \frac{67}{2080} = 0. \tag{29}$$

Solving equation (29) for τ gives the following values:

$$\tau = 1, \quad \tau = -0.0275947282.$$

Thus, by the definition of zero-stability, the method (13) is zero stable.

Similarly, the same procedure is used for $r = \frac{5}{8}$, the stability polynomial of (14) is obtained:

$$\begin{aligned}\beta_{\frac{5}{8}}(\tau, H) = \frac{13669}{11005}\tau^2 - \frac{27433}{22010}\tau^2 H - \frac{51193}{55025}\tau \\ + \frac{4368}{11005}\tau^2 H^2 - \frac{171799}{110050}\tau H - \frac{17152}{550025} \\ - \frac{2457}{11005}\tau H^2 - \frac{1536}{11005}H = 0.\end{aligned} \quad (30)$$

For zero stability, we set $H = 0$ in equation (30) to obtain the first characteristics polynomial as

$$\frac{13669}{11005}\tau^2 - \frac{51193}{55025}\tau - \frac{17152}{550025} = 0. \quad (31)$$

Solving equation (31) for τ gives the following values:

$$\tau = 1, \quad \tau = -0.2509620309.$$

Thus, by the definition of zero-stability, the method (14) is zero stable.

The boundary of the stability region of the methods (12), (13) and (14) are determined by substituting $\tau = e^{i\theta}$ into equations (26), (28) and (30). The graphs of the stability region for the methods (12), (13) and (14) using Maple 18 software are given as follows. Figure 1 illustrates the stability region of method (12) when $r = 1$.

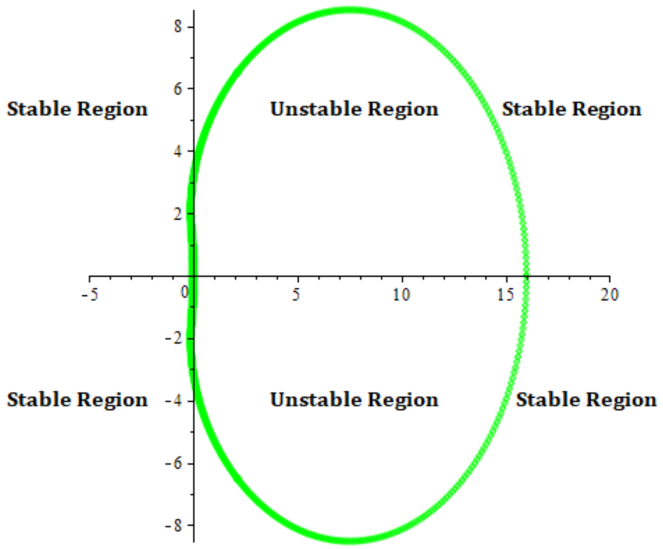

Figure 1. Stability region of the VSBBDF3 when $r = 1$.

Therefore, the stability region corresponds to the area outside the circular boundary, indicating that the method is A-stable, as the region encompasses the entire left half of the complex plane. Subsequently, the stability regions for methods (13) and (14) are presented in Figures 2 and 3, respectively. Figure 2 shows the stability region of method (13) when $r = 2$.

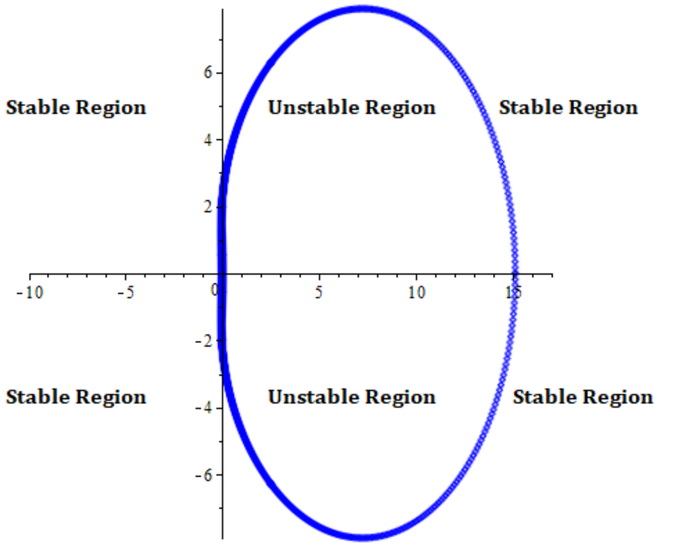

Figure 2. Stability region of the VSBBDF3 when $r = 2$.

Similarly, the stability region lies outside the circular boundary, indicating that the method is A-stable, as it encompasses the entire left half of the complex plane. The stability region for method (14) with $r = \frac{5}{8}$ is displayed in Figure 3.

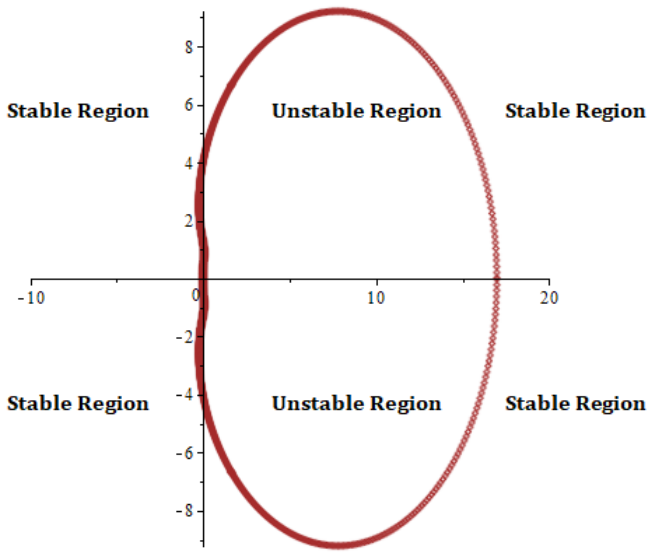

Figure 3. Stability region of the VSBBDF3 when $r = \frac{5}{8}$.

Analysis shows that the region of absolute stability for method (14) lies outside the circular boundary, indicating that it is A-stable, as the stability region encompasses the entire left half of the complex plane. A-stability is a crucial requirement for the reliable numerical solution of stiff IVPs. Since methods (12), (13), and (14) satisfy both zero-stability and A-stability criteria, they are well-suited for the numerical integration of stiff IVPs. The consistency and convergence properties of these methods are examined in the subsequent section.

5 Convergence of the Method

Convergence is a fundamental requirement for any acceptable linear multistep method (LMM). In this section, the convergence of the proposed methods is established using the classical Dahlquist Equivalence Theorem [4], which states that consistency and zero-stability are necessary and sufficient conditions for convergence.

Consistency determines the order and magnitude of the local truncation error, while zero-stability governs how errors are propagated during the numerical integration process. A linear multistep method is regarded as convergent if the numerical solution it generates approaches the exact solution as the step size $h \to 0$.

Thus, any method that fails to satisfy both consistency and zero-stability is non-convergent and must be discarded, as it is not suitable for the reliable numerical solution of differential equations.

Definition 8 (Consistency)**.** A LMM is said to be consistent if it has order at least one. Equivalently, a LMM is consistent if the following conditions are satisfied [24]:

$$\text{i.} \quad \sum_{k=0}^{j} \gamma_k = 0,$$

$$\text{ii.} \quad \sum_{k=0}^{j} k\gamma_k = \sum_{k=0}^{j} \Gamma_k.$$

Theorem 1 (Dahlquist Equivalence Theorem)**.** The necessary and sufficient conditions for a linear multi-step method (LMM) to be convergent are that it is consistent and zero stable [19, 25, 26].

Theorem 2. The third-order fully implicit variable step size super class of block backward differentiation formula (VSBBDF3) converges.

Proof. To show that the VSBBDF3 method converges, it is enough to show that the method is consistent and zero stable. We begin by showing that the method is consistent.

From what was established in the preceding section, we deduced that the order of the method is three which is greater than one. Let $\gamma_0, \gamma_1, \gamma_2, \gamma_3, \Gamma_0, \Gamma_1, \Gamma_2$ and Γ_3 be as previously defined. Then

$$\sum_{k=0}^{3} \gamma_k = \gamma_0 + \gamma_1 + \gamma_2 + \gamma_3$$
$$= \begin{bmatrix} -\frac{1}{15} \\ -\frac{1}{10} \end{bmatrix} + \begin{bmatrix} -\frac{11}{10} \\ \frac{9}{25} \end{bmatrix} + \begin{bmatrix} 1 \\ -\frac{63}{50} \end{bmatrix} + \begin{bmatrix} \frac{1}{6} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{32}$$

Thus, the first consistency condition in (i) is satisfied.

$$\sum_{k=0}^{3} k\gamma_k = (0)\gamma_0 + (1)\gamma_1 + (2)\gamma_2 + (3)\gamma_3$$
$$= (0)\begin{bmatrix} -\frac{1}{15} \\ \frac{1}{10} \end{bmatrix} + (1)\begin{bmatrix} -\frac{11}{10} \\ \frac{9}{25} \end{bmatrix} + (2)\begin{bmatrix} 1 \\ -\frac{63}{50} \end{bmatrix} + (3)\begin{bmatrix} \frac{1}{6} \\ 1 \end{bmatrix}$$
$$= \begin{bmatrix} \frac{7}{5} \\ \frac{21}{25} \end{bmatrix}. \tag{33}$$

and

$$\sum_{k=0}^{3} \Gamma_k = (\Gamma_0 + \Gamma_1 + \Gamma_2 + \Gamma_3)$$
$$= \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \frac{3}{5} \\ 0 \end{bmatrix} + \begin{bmatrix} \frac{4}{5} \\ \frac{9}{25} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{12}{25} \end{bmatrix} \right) = \begin{bmatrix} \frac{7}{5} \\ \frac{21}{25} \end{bmatrix}. \tag{34}$$

Hence,

$$\sum_{k=0}^{3} k\gamma_k = \sum_{k=0}^{3} \Gamma_k.$$

Thus, the second condition of consistency in (ii) is also satisfied.

The consistency conditions are therefore met. The same procedure is applied for the consistency of the methods (13) and (14). Hence, the VSBBDF3 methods are consistent. It has thus been shown that the VSBBDF3 is zero stable and satisfies the consistency conditions. Having these two conditions, consistency and zero stability, the VSBBDF3 is convergent. □

6 Implementation of the Method

Newton's iterative scheme is employed in the implementation of the method. The procedure is outlined as follows. Equations (12)-(14) can be written in the following general form:

$$v_{n+1} = \theta_1 v_{n+2} + \pi_1 h f_n + \pi_2 h f_{n+1} + \epsilon_1$$
$$v_{n+2} = \theta_2 v_{n+1} + \pi_3 h f_{n+1} + \pi_4 h f_{n+2} + \epsilon_2 \quad (35)$$

where ϵ_1 and ϵ_2 are the back values. The formulae in (35) can be cast in matrix form as:

$$(I - A_1)V = h(B_1 F_1 + B_2 F_2) + \xi, \quad (36)$$

where,

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad A_1 = \begin{bmatrix} 0 & \theta_1 \\ \theta_2 & 0 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 & \pi_1 \\ 0 & 0 \end{bmatrix},$$
$$B_2 = \begin{bmatrix} \pi_2 & 0 \\ \pi_3 & \pi_4 \end{bmatrix}, \quad V = \begin{bmatrix} v_{n+1} \\ v_{n+2} \end{bmatrix}, \quad F_1 = \begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix},$$
$$F_2 = \begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix}, \quad \xi = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \end{bmatrix}. \quad (2)$$

Let

$$F = (I - A_1)V - h(B_1 F_1 + B_2 F_2) - \xi = 0. \quad (37)$$

Newton's iteration for the VSBBDF3 method takes the form

$$V^{(i+1)}_{n+1,n+2} - V^{(i)}_{n+1,n+2} = -\left(F_j\left(V^{(i)}_{n+1,n+2}\right)\right)^{-1} \times \left(F_j\left(V^{(i)}_{n+1,n+2}\right)\right). \quad (38)$$

Equation (38) is equivalent to

$$V^{(i+1)}_{n+1,n+2} - V^{(i)}_{n+1,n+2} = -\left((I - A_1) - hB_1 \frac{\partial F_1}{\partial V}\left(V^{(i)}_{n+1,n+2}\right) - hB_2 \frac{\partial F_2}{\partial V}\left(V^{(i)}_{n+1,n+2}\right)\right)^{-1} \times \left((I - A_1)\left(V^{(i)}_{n+1,n+2}\right) - hB_1 F_1 - hB_2 F_2 - \xi\right). \quad (39)$$

For comparison purposes, the maximum error is evaluated using the developed algorithm. Let v_i and $v(u_i)$ denote the numerical and exact solutions of the initial value problem $v' = f(u, v)$, $v(a) = v_0$, respectively. The absolute error is defined as follows:

$$(\text{error}_i)_t = |(v_i)_t - (v(u_i))_t|, \quad (40)$$

The maximum error is defined by:

$$MAXE = \underbrace{\max}_{1 \le i \le T} \underbrace{(\max(error_i)_t)}_{1 \le i \le N}, \quad (41)$$

where T is the total number of steps and N is the number of equations.

Let $V^{(i+1)}_{n+1}$ denote the $(i+1)$th iterate and

$$E^{(i+1)}_{1,2} = V^{(i+1)}_{n+1,n+2} - V^{(i)}_{n+1,n+2}. \quad (42)$$

Equation (42) can be written as

$$E^{(i+1)}_{1,2} = \bar{A}^{-1}\bar{B}, \quad (43)$$

which is equivalent to

$$\bar{A}E^{(i+1)}_1 = \bar{B}. \quad (44)$$

where,

$$\bar{A} = (I - A_1) - hB_1 \frac{\partial F_1}{\partial V}\left(V^{(i)}_{n+1,n+2}\right) - hB_2 \frac{\partial F_2}{\partial V}\left(V^{(i)}_{n+1,n+2}\right),$$
$$\bar{B} = (I - A_1)\left(V^{(i)}_{n+1,n+2}\right) - hB_1 F_1 - hB_2 F_2 - \xi.$$

Newton's iteration would therefore be used to solve the system (44) for the different values of r.

$$\bar{A} = \begin{pmatrix} 1 - \pi_2 h \frac{\partial f_{n+1}}{\partial v_{n+1}} & -\theta_1 \\ -\theta_2 - \pi_3 h \frac{\partial f_{n+1}}{\partial v_{n+1}} & 1 - \pi_4 h \frac{\partial f_{n+2}}{\partial v_{n+2}} \end{pmatrix},$$

$$\bar{B} = \begin{pmatrix} -v^i_{n+1} + \theta_1 v^i_{n+2} + \pi_1 h f^i_n + \pi_2 h f^i_{n+1} + \epsilon_1 \\ -v^i_{n+2} + \theta_2 v^i_{n+1} + \pi_3 h f^i_{n+1} + \pi_4 h f^i_{n+2} + \epsilon_2 \end{pmatrix}.$$

Specifically, when $r = 1$:

$$\bar{A} = \begin{pmatrix} 1 - \frac{4}{5} h \frac{\partial f_{n+1}}{\partial v_{n+1}} & \frac{1}{6} \\ \frac{63}{50} - \frac{9}{25} h \frac{\partial f_{n+1}}{\partial v_{n+1}} & 1 - \frac{12}{25} h \frac{\partial f_{n+2}}{\partial v_{n+2}} \end{pmatrix},$$

$$\bar{B} = \begin{pmatrix} -v^i_{n+1} - \frac{1}{6} v^i_{n+2} + \frac{3}{5} h f^i_n + \frac{4}{5} h f^i_{n+1} + \frac{1}{15} v_{n-1} + \frac{11}{10} v_n \\ -v^i_{n+2} + \frac{63}{50} v^i_{n+1} + \frac{9}{25} h f^i_{n+1} + \frac{12}{25} h f^i_{n+2} + \frac{1}{10} v_{n-1} - \frac{9}{25} v_n \end{pmatrix}.$$

When $r = 2$:

$$\bar{A} = \begin{pmatrix} 1 - \frac{3}{4} h \frac{\partial f_{n+1}}{\partial v_{n+1}} & \frac{9}{64} \\ -\frac{232}{195} - \frac{24}{65} h \frac{\partial f_{n+1}}{\partial v_{n+1}} & 1 - \frac{32}{65} h \frac{\partial f_{n+2}}{\partial v_{n+2}} \end{pmatrix},$$

$$\bar{B} = \begin{pmatrix} -v^i_{n+1} - \frac{9}{64} v^i_{n+2} + \frac{9}{16} h f^i_n + \frac{3}{4} h f^i_{n+1} + \frac{1}{64} v_{n-1} + \frac{9}{8} v_n \\ -v^i_{n+2} + \frac{232}{195} v^i_{n+1} + \frac{24}{65} h f^i_{n+1} + \frac{32}{65} h f^i_{n+2} + \frac{1}{39} v_{n-1} - \frac{14}{65} v_n \end{pmatrix}.$$

When $r = \frac{5}{8}$:

$$\bar{A} = \begin{pmatrix} 1 - \frac{26}{31} h \frac{\partial f_{n+1}}{\partial v_{n+1}} & \frac{481}{2604} \\ -\frac{6048}{4615} - \frac{126}{355} h \frac{\partial f_{n+1}}{\partial v_{n+1}} & 1 - \frac{168}{355} h \frac{\partial f_{n+2}}{\partial v_{n+2}} \end{pmatrix},$$

$$\bar{B} = \begin{pmatrix} -v^i_{n+1} - \frac{481}{2604} v^i_{n+2} + \frac{39}{62} h f^i_n + \frac{26}{31} h f^i_{n+1} + \frac{512}{3255} v_{n-1} + \frac{637}{620} v_n \\ -v^i_{n+2} + \frac{6048}{4615} v^i_{n+1} + \frac{126}{355} h f^i_{n+1} + \frac{168}{355} h f^i_{n+2} + \frac{1024}{4615} v_{n-1} - \frac{189}{355} v_n \end{pmatrix}.$$

The proposed VSBBDF3 method is implemented using a computer program developed in the C programming language.

6.1 Local truncation error (LTE) and Step Size Selection

To regulate the integration process, three step-size control procedures are employed. At the initial stage of the computation, an appropriate starting step size is selected. The corresponding local truncation error is then evaluated using

$$\mathrm{LTE}_r = \left| v^{(p+1)}_{n+2} - v^{(p)}_{n+2} \right|. \quad (45)$$

where $v^{(p+1)}_{n+2}$ corresponds to the formula of order $p+1$ while $v^{(p)}_{n+2}$ refers to the formula of order p. The computed LTEs serve as a measure of accuracy and provide the basis for either accepting the current step or adjusting the step size to maintain the desired error tolerance.

For the VSBBDF3 Method, the LTEs are given by

i. When $r = 1$

$$v^{(3)}_{n+2} = \frac{1}{8} v_{n-1} - \frac{1}{2} v_n + \frac{11}{8} v_{n+1} + \frac{1}{2} h f_{n+2} + \frac{1}{4} h f_{n+1}$$
$$v^{(2)}_{n+2} = -\frac{1}{7} v_n + \frac{8}{7} v_{n+1} + \frac{4}{7} h f_{n+2} - \frac{2}{7} h f_{n+1}$$
$$\mathrm{LTE}_1 = \frac{1}{8} v_{n-1} - \frac{5}{14} v_n + \frac{13}{56} v_{n+1} - \frac{1}{14} h f_{n+2} + \frac{15}{28} h f_{n+1} \quad (46)$$

ii. When $r = 2$

$$v^{(3)}_{n+2} = \frac{1}{31} v_{n-1} - \frac{10}{31} v_n + \frac{40}{31} v_{n+1} + \frac{16}{31} h f_{n+2} - \frac{8}{31} h f_{n+1}$$
$$v^{(2)}_{n+2} = -\frac{1}{7} v_n + \frac{8}{7} v_{n+1} + \frac{4}{7} h f_{n+2} - \frac{2}{7} h f_{n+1}$$
$$\mathrm{LTE}_2 = \frac{1}{31} v_{n-1} - \frac{39}{217} v_n + \frac{32}{217} v_{n+1} - \frac{12}{217} h f_{n+2} + \frac{6}{217} h f_{n+1} \quad (47)$$

iii. When $r = \frac{5}{8}$

$$v^{(3)}_{n+2} = \frac{1024}{3705} v_{n-1} - \frac{203}{285} v_n + \frac{56}{39} v_{n+1} + \frac{28}{57} h f_{n+2} + \frac{14}{57} h f_{n+1}$$
$$v^{(2)}_{n+2} = -\frac{1}{7} v_n + \frac{8}{7} v_{n+1} + \frac{4}{7} h f_{n+2} - \frac{2}{7} h f_{n+1}$$
$$\mathrm{LTE}_5 = \frac{1024}{3705} v_{n-1} - \frac{1136}{1995} v_n + \frac{80}{273} v_{n+1} - \frac{32}{399} h f_{n+2} + \frac{212}{399} h f_{n+1} \quad (48)$$

The step-size control procedure can be summarized as follows:

1. If the computed local truncation error satisfies the tolerance requirement, that is LTE $\leq$ TOL, the step is accepted and the previous block step size is retained (*corresponding to* $r = 1$). A new step size is then estimated using
$$h_{\mathrm{new}} = c \times h_{\mathrm{old}} \times \left(\frac{\mathrm{TOL}}{\mathrm{LTE}} \right)^{1/p}$$
where p is the order of the method and $c = 0.7$ is the safety factor.
2. To avoid an excessive increase in the step length, if $h_{\mathrm{new}} > 1.6 \times h_{\mathrm{old}}$, then the step size is restricted to $h = 1.6 \times h_{\mathrm{old}}$ which corresponds to the case $r = 5/8$.
3. If the local truncation error exceeds the prescribed tolerance, that is LTE $>$ TOL, the step is rejected and the step size is reduced by half, $h = \frac{1}{2} h_{\mathrm{old}}$ which is equivalent to the case $r = 2$.

7 Test problems and Numerical results

To evaluate the performance and efficiency of the fully implicit variable step-size superclass of the third-order block backward differentiation formula, it is tested on selected systems of stiff initial value problems as follows:

Problem 1 (Highly Stiff System)

The well-known Gear's Problem, adopted from [27, 28], is given by

$$v_1' = 998 v_1 + 1998 v_2, \quad v_1(0) = 1, \quad 0 \leq u \leq 20,$$
$$v_2' = -999 v_1 - 1999 v_2, \quad v_2(0) = 0.$$

The exact solutions are

$$v_1(u) = 2e^{-u} - e^{-1000u},$$
$$v_2(u) = -e^{-u} + e^{-1000u}.$$

The corresponding eigenvalues of the Jacobian matrix are $\lambda_1 = -1$ and $\lambda_2 = -1000$. Since there is a large disparity in the eigenvalues, this problem is classified as highly stiff.

Problem 2 (Highly Stiff and Oscillatory System)

This stiff system, obtained from [29], is defined by

$$v_1' = -20 v_1 - 0.25 v_2 - 19.75 v_3, \; v_1(0) = 1, \; 0 \leq u \leq 10,$$
$$v_2' = 20 v_1 - 20.25 v_2 + 0.25 v_3, \; v_2(0) = 0,$$
$$v_3' = 20 v_1 - 19.75 v_2 - 0.25 v_3, \; v_3(0) = -1.$$

The exact solutions are

$$v_1(u) = 0.5\left(e^{-0.5u} + e^{-20u}(\cos 20u + \sin 20u)\right),$$
$$v_2(u) = 0.5\left(e^{-0.5u} - e^{-20u}(\cos 20u - \sin 20u)\right),$$
$$v_3(u) = -0.5\left(e^{-0.5u} + e^{-20u}(\cos 20u - \sin 20u)\right).$$

The eigenvalues are $\lambda_1 = -0.5$, $\lambda_2 = -20 + 20i$, and $\lambda_3 = -20 - 20i$. The presence of complex conjugate eigenvalues indicates oscillatory behavior, while the large negative real parts imply stiffness; hence, the problem is highly stiff and oscillatory.

Problem 3 (Highly Stiff Nonlinear System)

A nonlinear stiff reaction kinetics problem taken from [30, 31] is given by

$$v_1' = -1002v_1 + 1000v_2^2,$$
$$v_2' = v_1 - v_2(1 + v_2),$$
$$v_1(0) = 1, \quad 0 \le u \le 20,$$
$$v_2(0) = 1.$$

The exact solutions are

$$v_1(u) = e^{-u},$$
$$v_2(u) = e^{-u}.$$

The corresponding eigenvalue is $\lambda = -1002$. Due to the large negative eigenvalue, this system is classified as highly stiff (nonlinear).

Table 3. The notations used in the following tables are defined as:

SYMBOL	DESCRIPTION
TOL	Tolerance value used
METHOD	Method used
TRS	Number of rejected steps
TAS	Number of accepted steps
TS	Total number of steps
MAXE	Maximum absolute error
ODE15s	Variable-Step, Variable-Order Implicit Solver
ODE23s	Variable-Step Modified Rosenbrock Method
NBDF	Third Derivative Non-Block Method for Stiff ODEs [32]
VSBBDF	Fourth Order Variable Step-Size Block Backward Differentiation Formula [31]
VSBBDF3	Proposed Fully implicit third-order Variable Step-Size Superclass of Block Backward Differentiation Formula

The test problems were solved using the built-in Matlab ODE15s and ODE23s solvers, classical Non-Block Backward Differentiation Formula (NBDF) [32], the fourth order Variable Step-Size Block Backward Differentiation Formula (VSBBDF) [31], and our proposed fully implicit third-order Variable Step-Size Superclass of Block Backward Differentiation Formula (VSBBDF3).

Table 4. Numerical Result for Problem 1.

TOL	METHOD	TRS	TAS	TS	MAXE
	NBDF	13	58	71	3.52770E-001
	ODE15s	0	37	37	1.76000E-002
10^{-2}	ODE23s	0	22	22	7.31000E-003
	VSBBDF	0	54	54	2.92585E-004
	VSBBDF3	0	34	34	9.01240E-005
	NBDF	18	101	119	1.12690E-003
	ODE15s	0	89	89	1.86590E-004
10^{-4}	ODE23s	0	67	67	3.68370E-004
	VSBBDF	0	73	73	4.13979E-005
	VSBBDF3	0	62	62	9.23058E-007
	NBDF	22	165	187	6.80010E-006
	ODE15s	0	167	167	3.95690E-006
10^{-6}	ODE23s	0	287	287	1.70390E-005
	VSBBDF	0	122	122	2.03559E-006
	VSBBDF3	0	84	84	9.25266E-009

To evaluate and compare the accuracy and computational efficiency of these methods, the maximum absolute errors corresponding to various tolerance values (TOL) are presented for each problem. Furthermore, the numbers of rejected steps, accepted steps, and total integration steps used in solving each problem are also reported. The notations used in the following tables are defined in Table 3. The numerical results for Problems 1, 2, and 3 are presented in Tables 4, 5, and 6, respectively.

This approach ensures that the comparison is based on established benchmark results and is not affected by differences in implementation, thereby providing a fair and reliable evaluation of the proposed method.

To provide a visual representation of the performance of the method, the graphs of $\log_{10}(\text{MAXE})$ against TOL are plotted in Matlab software environment. The accuracy graph for Problem 1 is presented in Figure 4.

Similarly, the accuracy graphs for Problems 2 and 3 are shown in Figures 5 and 6, respectively.

Table 5. Numerical Result for Problem 2.

TOL	METHOD	TRS	TAS	TS	MAXE
	NBDF	8	48	56	4.92270E-002
	ODE15s	0	33	33	1.09000E-002
10^{-2}	ODE23s	0	20	20	6.40000E-003
	VSBBDF	0	46	46	4.09523E-003
	VSBBDF3	0	32	32	4.75020E-005
	NBDF	15	100	115	5.96300E-004
	ODE15s	0	70	70	8.55060E-005
10^{-4}	ODE23s	0	54	54	6.97740E-005
	VSBBDF	0	66	66	4.10262E-004
	VSBBDF3	0	58	58	4.97203E-007
	NBDF	19	167	186	1.04720E-005
	ODE15s	1	139	140	4.00590E-006
10^{-6}	ODE23s	0	232	232	1.70230E-005
	VSBBDF	1	117	118	2.31132E-006
	VSBBDF3	0	77	77	5.65481E-009

Table 6. Numerical Result for Problem 3.

TOL	METHOD	TRS	TAS	TS	MAXE
	NBDF	8	48	56	1.30840E-001
	ODE15s	0	29	29	5.20000E-003
10^{-2}	ODE23s	0	25	25	1.10000E-003
	VSBBDF	0	27	27	1.07390E-004
	VSBBDF3	0	21	21	2.31779E-004
	NBDF	17	84	101	1.19000E-003
	ODE15s	0	55	55	8.55060E-005
10^{-4}	ODE23s	0	118	118	6.97740E-005
	VSBBDF	0	47	47	5.48200E-006
	VSBBDF3	0	44	44	1.55132E-006
	NBDF	20	123	143	6.59590E-006
	ODE15s	0	197	197	1.07900E-006
10^{-6}	ODE23s	0	773	773	2.80810E-006
	VSBBDF	0	104	104	2.24940E-007
	VSBBDF3	0	63	63	3.05877E-009

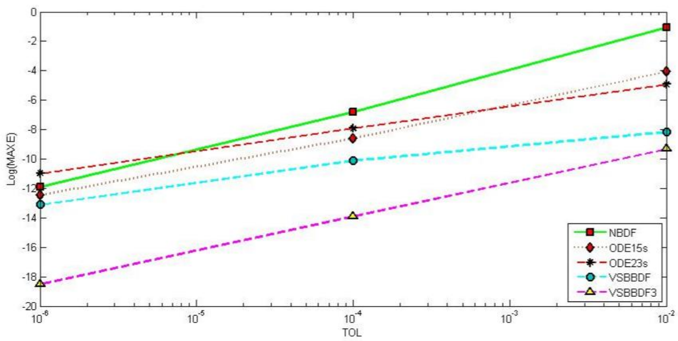

Figure 4. Accuracy Graph for Problem 1.

8 Discussion of Results

The numerical performance of the proposed and existing methods is evaluated using three stiff IVPs

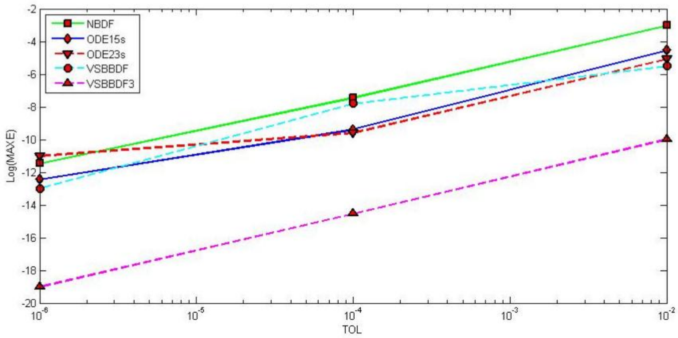

Figure 5. Accuracy Graph for Problem 2.

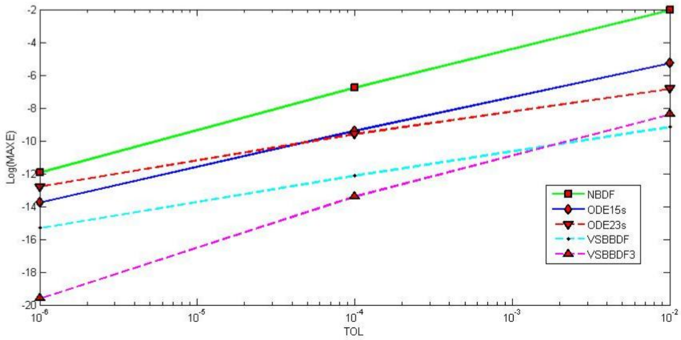

Figure 6. Accuracy Graph for Problem 3.

under varying tolerance levels. The comparison is based on computational effort, measured through the number of successful steps, rejected steps, and total steps, as well as accuracy assessed via the maximum absolute error.

The results consistently show that all methods improve in accuracy as the tolerance is tightened. This behavior is in agreement with theoretical expectations, since stricter tolerance levels enforce stronger local error control and consequently reduce the accumulation of global discretization errors. Across all test problems, the proposed VSBBDF3 method demonstrates superior accuracy in most cases, particularly at tighter tolerance levels from the literature, as well as the built-in MATLAB ODE solvers. This indicates its enhanced capability for handling stiffness and controlling numerical error effectively.

In terms of computational efficiency, the classical solvers generally exhibit increased computational cost as the stiffness of the problems becomes more pronounced. This reflects their reduced efficiency in dealing with highly stiff systems. The NBDF method exhibits more frequent step rejections, reflecting greater sensitivity to stiffness, while the VSBBDF method maintains better stability, though both achieve comparatively lower accuracy than the proposed VSBBDF3. In contrast, the proposed VSBBDF3 method achieves a more favorable balance between accuracy and computational effort. Notably, it maintains high accuracy without a corresponding increase in computational cost, demonstrating its efficiency advantage over the existing methods.

With respect to robustness, the proposed VSBBDF3 method exhibits consistent stability across all test problems and tolerance levels. The existing methods remain stable in most cases; however, their performance is generally more sensitive to increasing stiffness, particularly in terms of efficiency and accuracy. The classical built-in Matlab ODE solvers, in particular, show noticeable degradation in performance as stiffness increases, reflected in higher computational effort and reduced accuracy. The NBDF and VSBBDF methods perform better than the ODE15s and ODE23s but are still outperformed by the proposed method.

Overall, the numerical experiments clearly demonstrate that the VSBBDF3 method is the most accurate and reliable scheme among all the tested methods. It consistently provides improved accuracy while maintaining computational efficiency, making it particularly well suited for stiff ODEs. The results therefore confirm that the proposed VSBBDF3 method offers clear advantages over existing methods from the literature and classical Matlab ODE solvers in terms of accuracy, stability, and computational efficiency.

9 Conclusion

In this paper, a third-order variable step-size superclass of the block backward differentiation formula, denoted as VSBBDF3, has been successfully developed for the numerical simulation of nonlinear stiff dynamical systems governed by ordinary differential equations, with direct relevance to applications in dynamics, control engineering, and bifurcation analysis. The proposed method satisfies the fundamental requirements of consistency, zero-stability, and convergence, and is further shown to be A-stable, thereby confirming its theoretical soundness and suitability for stiff problems.

The performance of the method has been validated through extensive numerical experiments on a range of stiff linear, oscillatory, and nonlinear test problems. The results demonstrate that the proposed VSBBDF3 method is both highly accurate and computationally efficient. When compared with existing methods from the literature, including the non-block third derivative method (NBDF), the variable step-size block backward differentiation formula (VSBBDF), and the built-in MATLAB ODE solvers, the proposed method consistently delivers improved accuracy while maintaining competitive computational cost.

Furthermore, the results highlight the effectiveness of combining block formulation with adaptive step-size control. Although VSBBDF also incorporates these features, the proposed VSBBDF3 method exhibits superior performance, indicating that the structured superclass formulation, together with dynamic step-size adaptation, provides a more efficient balance between numerical accuracy and computational effort.

Overall, the proposed VSBBDF3 method constitutes a robust, reliable, and efficient numerical scheme for the integration of stiff ordinary differential equations. Its strong performance across a variety of test problems suggests that it is well suited for practical applications in scientific and engineering computations.

Data Availability Statement

Data will be made available on request.

Funding

This work was supported without any funding.

Conflicts of Interest

The authors declare no conflicts of interest.

AI Use Statement

The authors declare that no generative AI was used in the preparation of this manuscript.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Hairer, E., Wanner, G., & Nørsett, S. P. (1993). *Solving ordinary differential equations I: Nonstiff problems*. Berlin, Heidelberg: Springer Berlin Heidelberg. [CrossRef]

[2] Shampine, L. F., & Gear, C. W. (1979). A user's view of solving stiff ordinary differential equations. *SIAM review, 21*(1), 1-17. [CrossRef]

[3] Willoughby, R. A. (1974). International Symposium on Stiff Differential Systems. In *Stiff Differential Systems* (pp. 1-19). Boston, MA: Springer US. [CrossRef]

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

[5] Gear, C. W. (1971). *Numerical initial value problems in ordinary differential equations*. Prentice-Hall. https://dl.acm.org/doi/abs/10.5555/540426

[6] Butcher, J. C. (2016). *Numerical methods for ordinary differential equations* (3rd ed.). John Wiley & Sons. [CrossRef]

[7] Ibrahim, Z. B., Othman, K. I., & Suleiman, M. (2007). Implicit r-point block backward differentiation formula for solving first-order stiff ODEs. *Applied Mathematics and Computation, 186*(1), 558-565. [CrossRef]

[8] Musa, H., & Alhassan, B. (2025, December). Fully implicit 2-point super class of block extended backward differentiation formula for solving stiff ordinary differential equations. In *American Institute of Physics Conference Series* (Vol. 3338, No. 1, p. 040002). [CrossRef]

[9] Akinfenwa, O. A., Jator, S. N., & Yao, N. M. (2013). Continuous block backward differentiation formula for solving stiff ordinary differential equations. *Computers & Mathematics with Applications, 65*(7), 996-1005. [CrossRef]

[10] Dahlquist, G. G. (1963). A special stability problem for linear multistep methods. *BIT Numerical Mathematics, 3*(1), 27-43. [CrossRef]

[11] Ibrahim, Z. B., & Nasarudin, A. A. (2020). A class of hybrid multistep block methods with a–stability for the numerical solution of stiff ordinary differential equations. Mathematics, 8(6), 914. [CrossRef]

[12] Ibrahim, Z. B., Othman, K. I., & Suleiman, M. (2007). Variable step block backward differentiation formula for solving first order stiff ODEs. *Proceedings of the World Congress on Engineering, II*, 785-789. https://www.iaeng.org/publication/WCE2007/WCE2007_pp785-789.pdf

[13] Soomro, H., Daud, H., & Zainuddin, N. (2021, July). Convergence of the 3-point block backward differentiation formulas with off-step point for stiff ODEs. In *Journal of Physics: Conference Series* (Vol. 1943, No. 1, p. 012137). IOP Publishing. [CrossRef]

[14] Soomro, H., Zainuddin, N., Daud, H., Sunday, J., Jamaludin, N., Abdullah, A., ... & Kadir, E. A. (2023). 3-Point block backward differentiation formula with an off-step point for the solutions of stiff chemical reaction problems. *Journal of Mathematical Chemistry, 61*(1), 75-97. [CrossRef]

[15] Vijitha-Kumara, K. H. Y. (1985). *Variable stepsize variable order multistep methods for stiff ordinary differential equations*. Iowa State University. [CrossRef]

[16] Cash, J. R. (1980). On the integration of stiff systems of ODEs using extended backward differentiation formulae. *Numerische Mathematik, 34*(3), 235-246. [CrossRef]

[17] Calvo, M., Montijano, J. I., & Rández, L. (1993). A_0-stability of variable stepsize BDF methods. *Journal of Computational and Applied Mathematics, 45*(1–2), 29–39. [CrossRef]

[18] Alhassan, B., Musa, H., & Naghmeh, A. (2022). Convergence and order of the 2-point diagonally implicit block backward differentiation formula with two off-step points. *UMYU Scientifica, 1*(2), 30-38. [CrossRef]

[19] Dahlquist, G. (1956). Convergence and stability in the numerical integration of ordinary differential equations. *Mathematica Scandinavica*, 33-53. [CrossRef]

[20] Martín-Vaquero, J., & Vigo-Aguiar, J. (2007). Adapted BDF algorithms: higher-order methods and their stability. *Journal of Scientific Computing, 32*(2), 287-313. [CrossRef]

[21] Ola Fatunla, S. (1991). Block methods for second order ODEs. *International journal of computer mathematics, 41*(1-2), 55-63. [CrossRef]

[22] Okuonghae, R. I., & Ikhile, M. N. O. (2011). $A(\alpha)$-stable linear multistep methods for stiff IVPs in ODEs. *Acta Universitatis Palackianae Olomucensis Facultas Rerum Naturalium Mathematica, 50*(1), 73-90. https://dml.cz/handle/10338.dmlcz/141714

[23] Mohd Zawawi, I. S., Ibrahim, Z. B., & Othman, K. I. (2015). Derivation of diagonally implicit block backward differentiation formulas for solving stiff initial value problems. *Mathematical problems in engineering, 2015*(1), 179231. [CrossRef]

[24] Musa, H., & Alhassan, B. (2025). Super class of implicit extended backward differentiation formulae for the numerical integration of stiff initial value problems. *Computational Algorithms and Numerical Dimensions, 4*(1), 18-33. [CrossRef]

[25] Henrici, P. (1962). *Discrete variable methods in ordinary differential equations*. New York: Wiley.

[26] Wanner, G., & Hairer, E. (1996). *Solving ordinary differential equations II: Stiff and differential-algebraic problems* (2nd ed.). Berlin, Heidelberg: Springer. [CrossRef]

[27] Ibrahim, Z. B., Suleiman, M., & Othman, K. I. (2005). Fixed coefficients block backward differentiation formulas for the numerical solution of stiff ordinary differential equations. *European Journal of Scientific Research, 21*(3), 508-520. http://psasir.upm.edu.my/id/eprint/7019/1/114.pdf#page=138

[28] Soomro, H., Zainuddin, N., Daud, H., Sunday, J., Jamaludin, N., Abdullah, A., ... & Kadir, E. A. (2022). Variable step block hybrid method for stiff chemical kinetics problems. *Applied Sciences, 12*(9), 4484. [CrossRef]

[29] Yatim, S. A. M., Ibrahim, Z. B., Othman, K. I., & Suleiman, M. B. (2011). A quantitative comparison of numerical method for solving stiff ordinary differential equations. *Mathematical Problems in Engineering, 2011*(1), 193691. [CrossRef]

[30] Enright, W. H., Hull, T. E., & Lindberg, B. (1975). Comparing numerical methods for stiff systems of ODE: s. *BIT Numerical Mathematics*, *15*(1), 10-48. [CrossRef]

[31] Suleiman, M. B., Musa, H., Ismail, F., & Senu, N. (2013). A new variable step size block backward differentiation formula for solving stiff initial value problems. *International Journal of Computer Mathematics*, *90*(11), 2391-2408. [CrossRef]

[32] Adoghe, L. O., Omole, E. O., & Fadugba, S. E. (2022). Third derivative method for solving stiff system of ordinary differential equations. *International Journal of Mathematics in Operational Research*, *23*(3), 412-425. [CrossRef]

Najamuddeen Bala holds a Bachelor of Science (BSc) degree in Mathematics from Bayero University Kano, Kano State, Nigeria, as well as Master of Science (MSc) and Doctor of Philosophy (PhD) degrees in Mathematics from Umaru Musa Yar'adua University, Katsina, Katsina State, Nigeria with a specialization in Numerical Analysis. He is currently a lecturer in the Department of Statistics at Federal Polytechnic Kaura-Namoda, Zamfara State, Nigeria. His research interests focus on numerical analysis, with particular emphasis on designing, analyzing, and implementing efficient numerical methods for stiff ordinary differential equations. He has contributed to several publications in reputable peer-reviewed journals and is actively involved in teaching, supervising research, and developing advanced numerical algorithms for solving complex mathematical problems. (Email: Phdmth190179@students.umyu.edu.ng)

Hamisu Musa is a Professor of Numerical Analysis. He obtained his Bachelor of Science (B.Sc.) and Master of Science (M.Sc.) degrees in Mathematics from Bayero University, Kano, Nigeria, and earned his Ph.D. in Numerical Analysis from Universiti Putra Malaysia. He has successfully supervised numerous Ph.D. students to completion, including the first and third authors of this work, and continues to mentor additional Ph.D. candidates. In addition, he has supervised more than twenty M.Sc. Mathematics students to successful completion, contributing significantly to the development of advanced numerical methods for stiff Initial value problems. (Email: hamisu.musa@umyu.edu.ng)

Buhari Alhassan holds a National Certificate in Education (NCE) in Mathematics/Chemistry from Isah Kaita College of Education, Dutsinma, Katsina State, Nigeria as well as Bachelor of Science (BSc), Master of Science (MSc), and Doctor of Philosophy (PhD) degrees in Mathematics from Umaru Musa Yar'adua University, Katsina, Katsina State, Nigeria with a specialization in Numerical Analysis. He is currently an academic staff member in the Department of Mathematics and Statistics, College of Natural and Applied Sciences, Al-Qalam University, Katsina, Nigeria. His research interests focus on numerical analysis, with particular emphasis on the development, analysis, and implementation of numerical methods for ordinary and fractional differential equations. He has authored and co-authored several articles in reputable peer-reviewed journals and is actively engaged in teaching, supervising research, and advancing efficient numerical algorithms for solving complex mathematical problems. (Email: buharialhassan@auk.edu.ng)