JOURNAL OF SCIENCE AND TECHNOLOGY
ISSN: 2229-8460 e-ISSN: 2600-7924

JST

Vol. 17 No. 1 (2025) 11-22
https://publisher.uthm.edu.my/ojs/index.php/jst

An Improved Extended 2-Point Super Class of Block Backward Differentiation Formula for Solving Stiff Initial Value Problems

Adamu Abdulrahman[1]*, Musa Hamisu[2]

[1] *Department of Mathematics and Statistics, Faculty of Natural and Applied Sciences, Umaru Musa Yar'adua University, Katsina, NIGERIA*

*Corresponding Author: adamuaman@gmail.com
DOI: https://doi.org/10.30880/jst.2025.17.01.002

Article Info

Received: 2 February 2025
Accepted: 23 May 2025
Available online: 30 June 2025

Keywords

Stiff, initial value problems, zero-stability, A-stability, super class of block backward differentiation formula

Abstract

A novel, enhanced two-point block backward differentiation formula has been devised for solving stiff initial value problems of ordinary differential equations, boasting superior stability properties, including zero-stability and near A-stability. Comparative analysis reveals its exceptional accuracy and efficiency, outperforming some existing methods, thereby positioning it as a viable alternative solver for addressing stiff initial value problems.

1. Introduction

Ordinary differential equations (ODEs) are widely used to model various real-world physical phenomena, including those in engineering sciences, physics, biology, and chemistry. These equations describe the behavior of systems that change over time or space, and their solutions provide valuable insights into the underlying dynamics of the systems. In particular, initial value problems (IVPs) involving ODEs are commonly used to model a wide range of phenomena. An IVP involving an ODE is typically formulated as:

$$y' = f(x,y), \qquad y(x_0) = y_0, \qquad a \le x \le b \tag{1}$$

where y' represents the derivative of y with respect to x, $f(x,y)$ is a given function, x_0 is the initial point, and y_0 is the initial condition. The solution to this IVP provides the behavior of the system over time or space.
Stiff initial value problems (IVPs) are specific types of ODEs that are characterized by the presence of both fast and slow components in their solutions. This is supported by [1],[2],[3],[4],[5],[6],[7], and [8].

The presence of both fast and slow components in stiff IVPs makes them challenging to solve numerically. Traditional numerical methods, such as explicit Euler or Runge-Kutta methods can be inefficient or inaccurate when applied to stiff problems. These methods often require small time steps to capture the fast components, which can lead to computational inefficiency or instability.

To overcome these challenges, specialized numerical methods have been developed for solving stiff IVPs. These methods include implicit Euler or Runge-Kutta methods, Back ward Differentiation Formula (BDF), Rosenbrock methods, and exponential integrators, among others. These methods are designed to efficiently capture the slow components of the solution while accurately resolving the fast components.

Recent studies have focused on developing efficient and accurate numerical methods for solving stiff IVPs. These studies have explored various approaches, including the development of new numerical methods, the improvement of existing methods, and the application of stiff IVP solvers to real-world problems.

Consider the Super class of Block Backward Differentiation Formula (SBBDF) for solving stiff initial value problems [9]:

$$\sum_{j=0}^{3} \alpha_{j,i}\, y_{n+j-1} = h\beta_{k,i}(f_{n+k} - \rho f_{n+k-2}), \; k = i = 1, 2 \tag{2}$$

where, $k = i = 1$ represents the formula for the first point and $k = i = 2$ corresponds to the formula for the second point. This paper introduces a compelling and efficient concept to enhance the extended version of (2), resulting in the development of almost A-stable block method capable of addressing stiff IVPs. The method takes the following form:

$$\sum_{j=0}^{4} \alpha_{j,i}\, y_{n+j-2} = h\beta_{k,i}(f_{n+k} - \rho f_{n+k-2}), \; k = i = 1, 2 \tag{3}$$

The formula (3) computes two solution values per step concurrently in block. To maintain good stability property of the method, the parameter ρ is restricted to a value in the interval (-1,1) as in [9]. The aim of this paper is to develop a higher order 2-point implicit block method with better stability and convergence properties when compared with existing 1-point BDF and conventional 2-point BBDF methods found in the literature.

2. Mathematical Formulation of the Method

The formula (3) is derived through Taylor's series expansion about any point x_n by the help of the following definition of a linear difference operator associated with (3).

Definition 1: The general linear operator, L_i associated with (3) is defined as:

$$\begin{aligned} L_i[y(x_n), h]&: \alpha_{0,i} y_{n-2} + \alpha_{1,i} y_{n-1} + \alpha_{2,i} y_n + \alpha_{3,i} y_{n+1} + \alpha_{4,i} y_{n+2} - h\beta_{k,i}(f_{n+k} - \rho f_{n+k-2}) \\ &= 0, k = i = 1,2 \end{aligned} \tag{4}$$

To derive the coefficients of the first point, we let $k = i = 1$ and the linear difference operator in (4) becomes:

$$\begin{aligned} L_1[y(x_n), h] = {}& \alpha_{0,1} y(x_n - 2h) + \alpha_{1,1} y(x_n - h) + \alpha_{2,1} y(x_n) + \alpha_{3,1} y(x_n + h) \\ &+ \alpha_{4,1} y(x_n + 2h) - h\beta_{1,1}\big(f(x_n + h) - \rho f(x_n - h)\big) = 0 \end{aligned} \tag{5}$$

Expanding (5) using Taylor series about x_n and collecting like terms gives:

$$C_{0,1} y(x_n) + C_{1,1} h y'(x_n) + C_{2,1} h^2 y''(x_n) + C_{3,1} h^3 y'''(x_n) + \cdots = 0 \tag{6}$$

where

$$\left.\begin{aligned} C_{0,1} &= \alpha_{0,1} + \alpha_{1,1} + \alpha_{2,1} + \alpha_{3,1} + \alpha_{4,1} = 0 \\ C_{1,1} &= -2\alpha_{0,1} - \alpha_{1,1} + \alpha_{3,1} + 2\alpha_{4,1} - \beta_{1,1}(1 - \rho) = 0 \\ C_{2,1} &= 2\alpha_{0,1} + \tfrac{1}{2}\alpha_{1,1} + \tfrac{1}{2}\alpha_{3,1} + 2\alpha_{4,1} - \beta_{1,1}(1 + \rho) = 0 \\ C_{3,1} &= -\tfrac{4}{3}\alpha_{0,1} - \tfrac{1}{6}\alpha_{1,1} + \tfrac{1}{6}\alpha_{3,1} + \tfrac{4}{3}\alpha_{4,1} - \beta_{1,1}\left(\tfrac{1}{2} - \tfrac{\rho}{2}\right) = 0 \\ C_{4,1} &= \tfrac{2}{3}\alpha_{0,1} + \tfrac{1}{24}\alpha_{1,1} + \tfrac{1}{24}\alpha_{3,1} + \tfrac{2}{3}\alpha_{4,1} - \beta_{1,1}\left(\tfrac{1}{6} + \tfrac{\rho}{6}\right) = 0 \end{aligned}\right\} \tag{7}$$

Equation (7) is solved simultaneously in Maple software environment and by normalizing the coefficient of y_{n+1} to one, we obtain the following coefficient of the first point as given in the table below:

J. of Science and Technology Vol. 17 No. 1 (2025) 11-22

Table 1 *Coefficient of the first point*

$\alpha_{0,1}$	$\alpha_{1,1}$	$\alpha_{2,1}$	$\alpha_{3,1}$	$\alpha_{4,1}$	$\beta_{1,1}$
$\frac{-(1-3\rho)}{2(5+3\rho)}$	$\frac{-(-3-5\rho)}{5+3\rho}$	$\frac{-9(1+\rho)}{5+3\rho}$	1	$\frac{-(-3+\rho)}{2(5+3\rho)}$	$\frac{6}{5+3\rho}$

Putting these values of $\alpha_{j,i}$ and $\beta_{k,i}$ in equation (5), the first point is therefore obtained as:

$$y_{n+1} = \frac{(1-3\rho)}{2(5+3\rho)}y_{n-2} + \frac{(-3-5\rho)}{5+3\rho}y_{n-1} + \frac{9(1+\rho)}{5+3\rho}y_n + \frac{(-3+\rho)}{2(5+3\rho)}y_{n+2} + \frac{6h}{5+3\rho}f_{n+1} - \frac{6\rho h}{5+3\rho}f_{n-1} \quad (8)$$

Now, the same procedure is applied in the derivation of the second point to obtain:

$$y_{n+2} = \frac{(-3+\rho)}{25+\rho}y_{n-2} + \frac{8(2-\rho)}{25+\rho}y_{n-1} - \frac{36}{25+\rho}y_n + \frac{8(6+\rho)}{25+\rho}y_{n+1} + \frac{12h}{25+\rho}f_{n+2} - \frac{12\rho h}{25+\rho}f_n \quad (9)$$

Therefore, by combining the formulae in (8) and (9), the required improved extended 2-point super class of block backward differentiation formula is thus:

$$\left.\begin{aligned} y_{n+1} &= \frac{(1-3\rho)}{2(5+3\rho)}y_{n-2} + \frac{(-3-5\rho)}{5+3\rho}y_{n-1} + \frac{9(1+\rho)}{5+3\rho}y_n + \frac{(-3+\rho)}{2(5+3\rho)}y_{n+2} + \frac{6h}{5+3\rho}f_{n+1} - \frac{6\rho h}{5+3\rho}f_{n-1} \\ y_{n+2} &= \frac{(-3+\rho)}{25+\rho}y_{n-2} + \frac{8(2-\rho)}{25+\rho}y_{n-1} - \frac{36}{25+\rho}y_n + \frac{8(6+\rho)}{25+\rho}y_{n+1} + \frac{12h}{25+\rho}f_{n+2} - \frac{12\rho h}{25+\rho}f_n \end{aligned}\right\} \quad (10)$$

In this paper, we will denote the formulae (10) as IE2SBBDF.

According to recent studies, selecting the free parameter ρ within the range of (-1,1) is crucial for upholding the A-stability of numerical methods. This is supported by [10],[11],[12],[13],[14],[15],[16], and [17]. For demonstration purposes, we choose $\rho = -\frac{1}{2}$ in (10) to obtain:

$$\left.\begin{aligned} y_{n+1} &= \frac{5}{14}y_{n-2} - \frac{1}{7}y_{n-1} + \frac{9}{7}y_n - \frac{1}{2}y_{n+2} + \frac{12}{7}hf_{n+1} + \frac{6}{7}hf_{n-1} \\ y_{n+2} &= -\frac{1}{7}y_{n-2} + \frac{40}{49}y_{n-1} - \frac{72}{49}y_n + \frac{88}{49}y_{n+1} + \frac{24}{49}hf_{n+2} + \frac{12}{49}hf_n \end{aligned}\right\} \quad (11)$$

The order of the method (11) is computed using Maple application package and shown to be of order four with error constant which is given by:

$$E_5 = \begin{pmatrix} \frac{9}{70} \\ -\frac{26}{245} \end{pmatrix}$$

3. Stability of the Method

The stability properties of the IE2SBBDF (11) are determined through application of the standard linear test problem:

$$y' = \lambda y,\ \ \lambda < 0,\ \ \lambda \text{ is complex} \quad (12)$$

Substitute (12) into (11) to obtain:

$$\left.\begin{aligned} y_{n+1} &= \frac{5}{14}y_{n-2} - \frac{1}{7}y_{n-1} + \frac{9}{7}y_n - \frac{1}{2}y_{n+2} + \frac{12}{7}h\lambda y_{n+1} + \frac{6}{7}h\lambda y_{n-1} \\ y_{n+2} &= -\frac{1}{7}y_{n-2} + \frac{40}{49}y_{n-1} - \frac{72}{49}y_n + \frac{88}{49}y_{n+1} + \frac{24}{49}h\lambda y_{n+2} + \frac{12}{49}h\lambda y_n \end{aligned}\right\} \quad (13)$$

which is equivalent to:

$$\left.\begin{aligned} y_{n+1} - \frac{12}{7}h\lambda y_{n+1} + \frac{1}{2}y_{n+2} &= \frac{5}{14}y_{n-2} - \frac{1}{7}y_{n-1} + \frac{9}{7}y_n + \frac{6}{7}h\lambda y_{n-1} \\ -\frac{88}{49}y_{n+1} + y_{n+2} - \frac{24}{49}h\lambda y_{n+2} &= -\frac{1}{7}y_{n-2} + \frac{40}{49}y_{n-1} - \frac{72}{49}y_n + \frac{12}{49}h\lambda y_n \end{aligned}\right\} \quad (14)$$

The matrix representation of (14) is given by:

$$\begin{pmatrix} (1-\frac{12}{7}h\lambda) & \frac{1}{2} \\ -\frac{88}{49} & (1-\frac{24}{49}h\lambda) \end{pmatrix}\begin{pmatrix} y_{n+1} \\ y_{n+2} \end{pmatrix} = \begin{pmatrix} (-\frac{1}{7}+\frac{6}{7}h\lambda) & \frac{9}{7} \\ \frac{40}{49} & (-\frac{72}{49}+\frac{12}{49}h\lambda) \end{pmatrix}\begin{pmatrix} y_{n-1} \\ y_n \end{pmatrix} + \begin{pmatrix} 0 & \frac{5}{14} \\ 0 & -\frac{1}{7} \end{pmatrix}\begin{pmatrix} y_{n-3} \\ y_{n-2} \end{pmatrix} \quad (15)$$

Putting $\bar{h} = h\lambda$ in (15) gives

$$\begin{pmatrix} (1-\frac{12}{7}\bar{h}) & \frac{1}{2} \\ -\frac{88}{49} & (1-\frac{24}{49}\bar{h}) \end{pmatrix}\begin{pmatrix} y_{n+1} \\ y_{n+2} \end{pmatrix} = \begin{pmatrix} (-\frac{1}{7}+\frac{6}{7}\bar{h}) & \frac{9}{7} \\ \frac{40}{49} & (-\frac{72}{49}+\frac{12}{49}\bar{h}) \end{pmatrix}\begin{pmatrix} y_{n-1} \\ y_n \end{pmatrix} + \begin{pmatrix} 0 & \frac{5}{14} \\ 0 & -\frac{1}{7} \end{pmatrix}\begin{pmatrix} y_{n-3} \\ y_{n-2} \end{pmatrix} \quad (16)$$

Equation (16) is equivalent to

$$AY_m = BY_{m-1} + CY_{m-2}, \;\; n = 2m \quad (17)$$

Where $A = \begin{pmatrix} (1-\frac{12}{7}\bar{h}) & \frac{1}{2} \\ -\frac{88}{49} & (1-\frac{24}{49}\bar{h}) \end{pmatrix}$, $B = \begin{pmatrix} (-\frac{1}{7}+\frac{6}{7}\bar{h}) & \frac{9}{7} \\ \frac{40}{49} & (-\frac{72}{49}+\frac{12}{49}\bar{h}) \end{pmatrix}$, $C = \begin{pmatrix} 0 & \frac{5}{14} \\ 0 & -\frac{1}{7} \end{pmatrix}$,

$$Y_m = \begin{pmatrix} y_{n+1} \\ y_{n+2} \end{pmatrix} = \begin{pmatrix} y_{2m+1} \\ y_{2m+2} \end{pmatrix},$$

$$Y_{m-1} = \begin{pmatrix} y_{n-1} \\ y_n \end{pmatrix} = \begin{pmatrix} y_{2m-1} \\ y_{2m} \end{pmatrix} = \begin{pmatrix} y_{2(m-1)+1} \\ y_{2(m-1)+2} \end{pmatrix},$$

$$Y_{m-2} = \begin{pmatrix} y_{n-3} \\ y_{n-2} \end{pmatrix} = Y_{m-2} = \begin{pmatrix} y_{2m-3} \\ y_{2m-2} \end{pmatrix} = \begin{pmatrix} y_{2(m-1)-1} \\ y_{2(m-1)} \end{pmatrix},$$

To obtain the stability polynomial of the method, the following equation is computed:

$$\det(At^2 + Bt + C) = 0 \quad (18)$$

where det refers to the determinant.

Equation (18) is equivalent to the following stability polynomial of the method:

$$R(t,\bar{h}) = \frac{93}{49}t^4 - \frac{108}{49}t^4\bar{h} - \frac{99}{343}t^3 - \frac{1266}{343}t^3\bar{h} - \frac{459}{343}t^2 + \frac{288}{343}t^4\bar{h}^2 + \frac{288}{343}t^3\bar{h}^2 - \frac{528}{343}t^2\bar{h} - \frac{93}{343}t + \frac{72}{343}t^2\bar{h}^2 - \frac{6}{49}t\bar{h} = 0 \quad (19)$$

J. of Science and Technology Vol. 17 No. 1 (2025) 11-22

Setting $\bar{h} = \lambda h = 0$ in (19), we have

$$\frac{93}{49}t^4 - \frac{99}{343}t^3 - \frac{459}{343}t^2 - \frac{93}{343}t = 0 \quad (20)$$

Solving (20) for t, we obtain

$$t = 0, t = 1, t = -0.6160245712\,, t = -0.2319016960$$

Definition 2 (Zero stability)**:** The method (11) is considered as zero stable when all roots of the first characteristic polynomial have a modulus less than or equal to one, and any root with a modulus of one is simple [18]. Therefore, based on this definition, the method (11) is deemed zero stable.
Definition 3 (A-stability)**:** A numerical method is said to be A-stable if its absolute stability region covers the entire left-hand of the complex plane [19].
Definition 4 (Almost A-stability)**:** A method is termed almost A-stable if its stability region encompasses almost the whole of the left-hand complex plane [19].
The stability polynomial (19) of the method (11) is shown in the following figure:

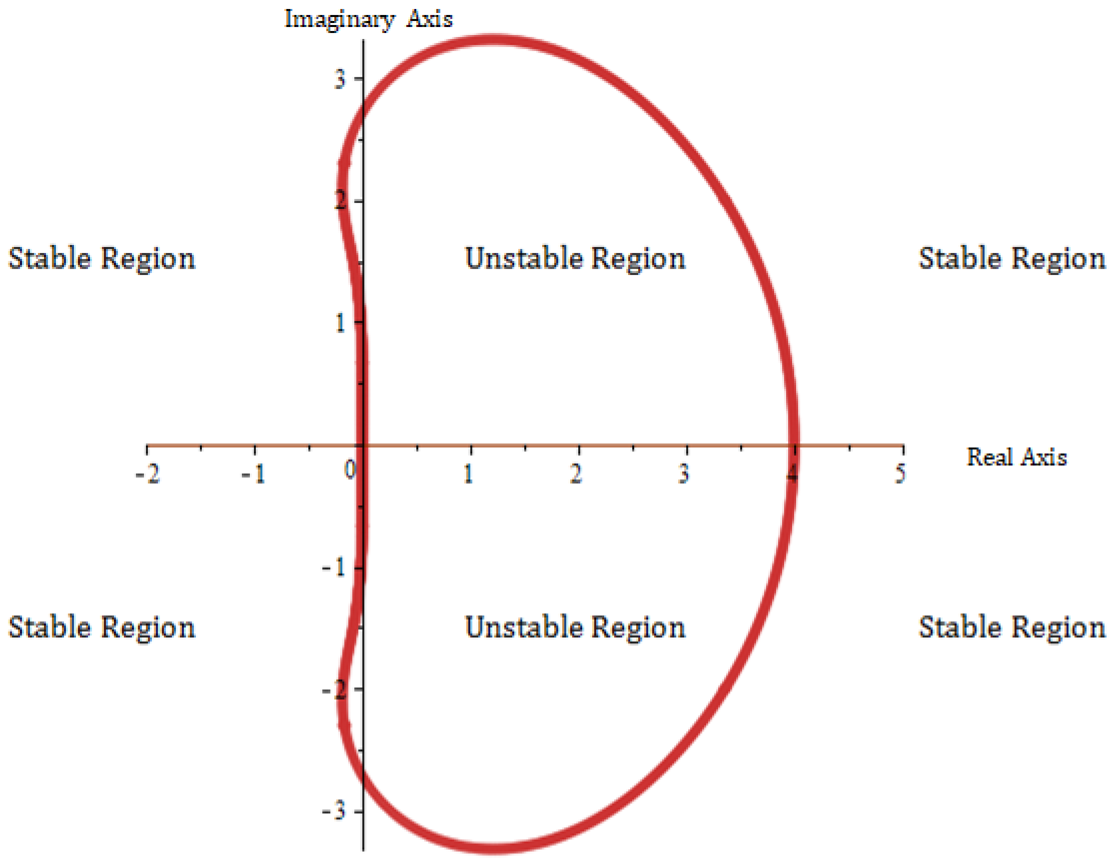

Fig. 1 *The stability region of the 21ESBBDF method*

The graph of the stability polynomial (19) depicted in Fig. 1 indicates that the stability region spans nearly the entire negative left-hand plane. This suggests that the method is almost A-stable and suitable for addressing stiff IVPs.

4. Implementation of the Method

The method incorporates Newton's iteration in its implementation, and the iteration details are provided below, starting with the definition of the error.

Definition 5: Lety_i and $y(x_i)$ be the approximate and exact solutions of (1) respectively. Then the absolute error is given by:

$$(error_i)_t = \left|(y_i)_t - \left(y(x_i)\right)_t\right| \tag{21}$$

The maximum error is given by:

$$MAXE = \underbrace{max}_{1\le i\le T}\left(\underbrace{max}_{1\le i\le T}(error_i)_t\right) \tag{22}$$

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

Define

$$\left.\begin{aligned} F_1 &= y_{n+1} + \frac{1}{2}y_{n+2} - \frac{12}{7}hf_{n+1} - \frac{6}{7}hf_{n-1} - \xi_1 \\ F_2 &= y_{n+2} - \frac{88}{49}y_{n+1} - \frac{24}{49}hf_{n+2} - \frac{12}{49}hf_n - \xi_2 \end{aligned}\right\} \tag{23}$$

where:

$$\left.\begin{aligned} \xi_1 &= \frac{5}{14}y_{n-2} - \frac{1}{7}y_{n-1} + \frac{9}{7}y_n \\ \xi_2 &= -\frac{1}{7}y_{n-2} + \frac{40}{49}y_{n-1} - \frac{72}{49}y_n \end{aligned}\right\} \tag{24}$$

Let $y_{n+j}^{(i+1)}$ denote the $(i+1)th$ iteration and

$$e_{n+j}^{(i+1)} = y_{n+j}^{(i+1)} - e_{n+j}^{(i)}\,,\quad j = 1,2 \tag{25}$$

The Newton's iteration for the IE2SBBDF takes the form

$$y_{n+j}^{(i+1)} - y_{n+j}^{(i)} = -\left[F_j'\left(y_{n+j}^{(i)}\right)\right]^{-1}\left[F_j\left(y_{n+j}^{(i)}\right)\right],\ j = 1,2 \tag{26}$$

which can be rewritten in the form:

$$\left[F_j'\left(y_{n+j}^{(i)}\right)\right]e_{n+j}^{(i+1)} = -\left[F_j\left(y_{n+j}^{(i)}\right)\right] \tag{27}$$

and in matrix form, equation (27) is equivalent to

$$\begin{pmatrix} 1 - \frac{12}{7}h\frac{\delta f_{n+1}}{\delta y_{n+1}} & \frac{1}{2} \\ -\frac{88}{49} & 1 - \frac{24}{49}h\frac{\delta f_{n+2}}{\delta y_{n+2}} \end{pmatrix}\begin{pmatrix} e_{n+1}^{(i+1)} \\ e_{n+2}^{(i+1)} \end{pmatrix} = \begin{pmatrix} -1 & -\frac{1}{2} \\ \frac{88}{49} & -1 \end{pmatrix}\begin{pmatrix} y_{n+1}^{(i)} \\ y_{n+2}^{(i)} \end{pmatrix} + h\begin{pmatrix} \frac{6}{7} & 0 \\ 0 & \frac{12}{49} \end{pmatrix}\begin{pmatrix} f_{n-1}^{(i)} \\ f_n^{(i)} \end{pmatrix} + h\begin{pmatrix} \frac{12}{7} & 0 \\ 0 & \frac{24}{49} \end{pmatrix}\begin{pmatrix} f_{n+1}^{(i)} \\ f_{n+2}^{(i)} \end{pmatrix} + \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} \tag{28}$$

Equation (28) is implemented using C programming language to get the desired numerical results.

5. Test Problems and Numerical Results

The following stiff problems are considered:

Problem 1:

$$y_1' = -1002y_1 + 1000y_2^2, \quad y_1(0) = 1, \quad 0 \le x \le 20,$$
$$y_2' = y_1 - y_2(1 + y_2), \quad y_2(0) = 1.$$

Exact Solution:

$$y_1(x) = e^{-2x},$$
$$y_2(x) = e^{-x}$$

Eigenvalues: -1 and -1002

Source: [13]

Problem 2:

$$y_1' = -100y_1 + 9.901y_2, \quad y_1(0) = 1, \quad 0 \le x \le 10,$$
$$y_2' = 0.1y_1 - y_2, \quad y_2(0) = 10.$$

Exact Solution:

$$y_1(x) = e^{-0.99x}$$
$$y_2(x) = 10e^{-0.99x}$$

Eigenvalues: -0.99 and -100.01

Source: [20]

Problem 3:

$$y_1' = -y_1 + 95y_2, \quad y_1(0) = 1, \quad 0 \le x \le 10.$$
$$y_2' = -y_1 - 97y_2, \quad y_2(0) = 1.$$

Exact Solution:

$$y_1(x) = \frac{1}{47}(95e^{-2x} - 48e^{-96x})$$
$$y_2(x) = \frac{1}{47}(48e^{-96x} - e^{-2x})$$

Eigenvalues: -2 and -96

Source: [21]

The problems were solved by the application of the developed method, the 1-point BDF and the 2-point BBDF with various step sizes (h). The following tables present details on the maximum error and computation time for each problem and the notations used in the tables are as follows:

h=step size;
1 BDF = 1-point BDF method;
2BBDF=2-point BBDF method;
IE2SBBDF= Improved extended 2-point super class of BBDF method;
NS=Total number of integration steps;
MAXE=Maximum Error;
Time=Computation time.

Table 2 *Numerical results for problem 1*

H	METHOD	NS	MAXE	TIME
10^{-2}	1BDF	2000	1.29417E+165	1.56500E-001
	2BBDF	1000	7.71189E+155	1.71600E-001
	IE2SBBDF	1000	5.80696E-003	1.99900E-001
10^{-3}	1BDF	20000	3.67821E-004	2.30400E-001
	2BBDF	10000	2.52592E+005	1.60100E-001
	IE2SBBDF	10000	1.08522E-004	1.36900E-001
10^{-4}	1BDF	200000	3.69038E-005	3.67400E-001
	2BBDF	100000	7.37885E-005	2.80700E-001
	IE2SBBDF	100000	1.09262E-005	5.07100E-001
10^{-5}	1BDF	2000000	3.69177E-006	1.211OOE+000
	2BBDF	1000000	7.38334E-006	1.19100E+000
	IE2SBBDF	1000000	1.09377E-006	2.90000E+000
10^{-6}	1BDF	20000000	3.69220E-007	9.76800E+000
	2BBDF	10000000	7.38360E-007	9.45700E+000
	IE2SBBDF	10000000	1.09400E-007	2.71400E+001

Table 3 *Numerical results for problem 2*

H	METHOD	NS	MAXE	TIME
10^{-2}	1BDF	1000	3.59051E-002	2.52275E-003
	2BBDF	500	7.11312E-002	2.81342E-003
	IE2SBBDF	500	1.02046E-002	3.21200E-002
10^{-3}	1BDF	10000	3.63689E-003	2.43456E-002
	2BBDF	5000	7.26689E-003	2.74121E-002
	IE2SBBDF	5000	1.07309E-003	5.04900E-002
10^{-4}	1BDF	100000	3.64150E-004	2.41299E-001
	2BBDF	50000	7.28230E-004	2.72916E-001
	IE2SBBDF	50000	1.07851E-004	1.69600E-001
10^{-5}	1BDF	1000000	3.64196E-005	2.39406E+000
	2BBDF	500000	7.28384E-005	2.72181E+000
	IE2SBBDF	500000	1.07905E-005	1.43300E+000
10^{-6}	1BDF	10000000	3.64219E-006	2.39058E+001
	2BBDF	5000000	7.28384E-006	2.72745E+001
	IE2SBBDF	5000000	1.07917E-006	1.32800E+001

Table 4 *Numerical results for problem 3*

H	METHOD	NS	MAXE	TIME
10^{-2}	1BDF	1000	2.22166E-002	4.03375E-003
	2BBDF	500	3.62349E+001	4.36325E-003
	IE2SBBDF	500	1.70437E-002	2.99700E-002
10^{-3}	1BDF	10000	3.08027E-002	3.89183E-002
	2BBDF	5000	5.62364E-002	4.22848E-002
	IE2SBBDF	5000	6.37905E-003	4.59800E-002
10^{-4}	1BDF	100000	3.55724E-003	3.84906E-001
	2BBDF	50000	7.04927E-003	4.19562E-001
	IE2SBBDF	50000	1.01227E-003	1.79900E-001
10^{-5}	1BDF	1000000	3.60179E-004	3.84739E+000
	2BBDF	500000	7.19695E-004	4.18440E+000
	IE2SBBDF	500000	1.06286E-004	1.47500E+000
10^{-6}	1BDF	10000000	3.60621E-005	3.82909E+001
	2BBDF	5000000	7.21175E-005	4.18912E+001
	IE2SBBDF	5000000	1.06807E-005	1.45600E+001

The proposed method demonstrates strong agreement between computed and exact solutions, with its maximum absolute error signaling accuracy. When compared to the 1BDF and 2BBDF methods, the newly developed approach outperforms the other two in terms of accuracy. In the realm of total integration steps, both IE2SBBDF and 2BBDF methods require only half the steps compared to the 1BDF method. Despite involving additional function evaluations, the new method remains competitive in computation time. A graphical representation below illustrates the accuracy of these methods, plotting $LOG_{10}(MAXE)$ against h for each tested problem.

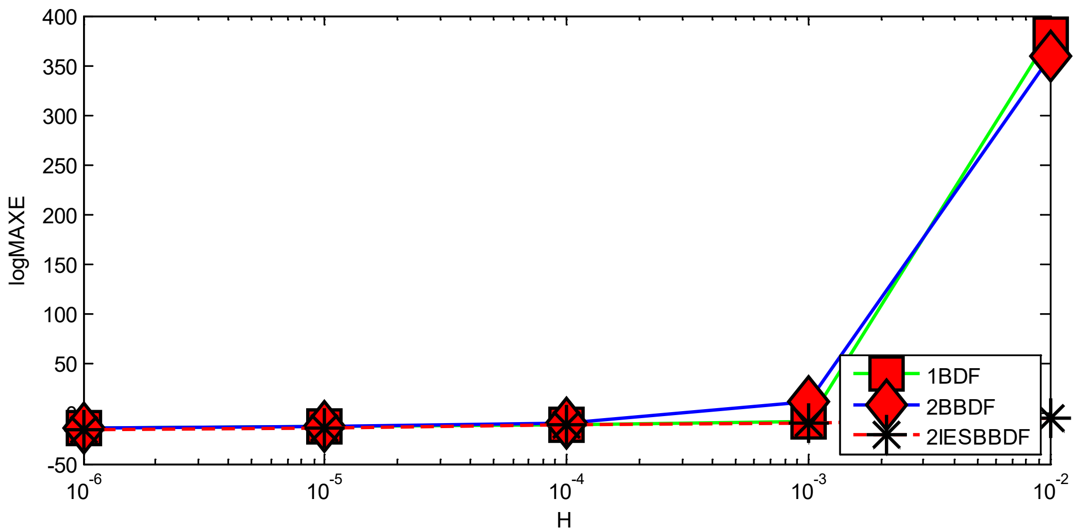

Fig. 2 *Graph of* $Log_{10}MAXE$ *against H for problem 1*

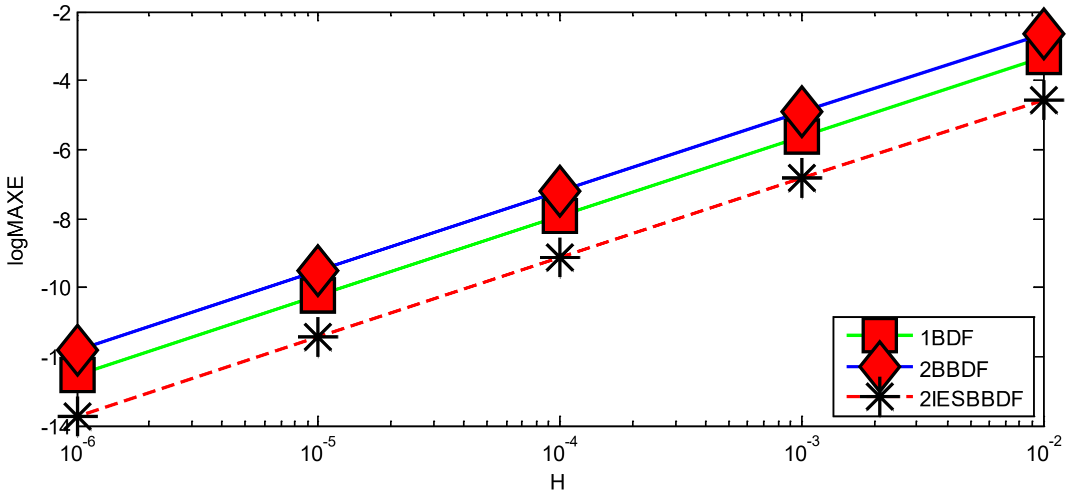

Fig. 3 *Graph of $Log_{10}MAXE$ against H for problem 2*

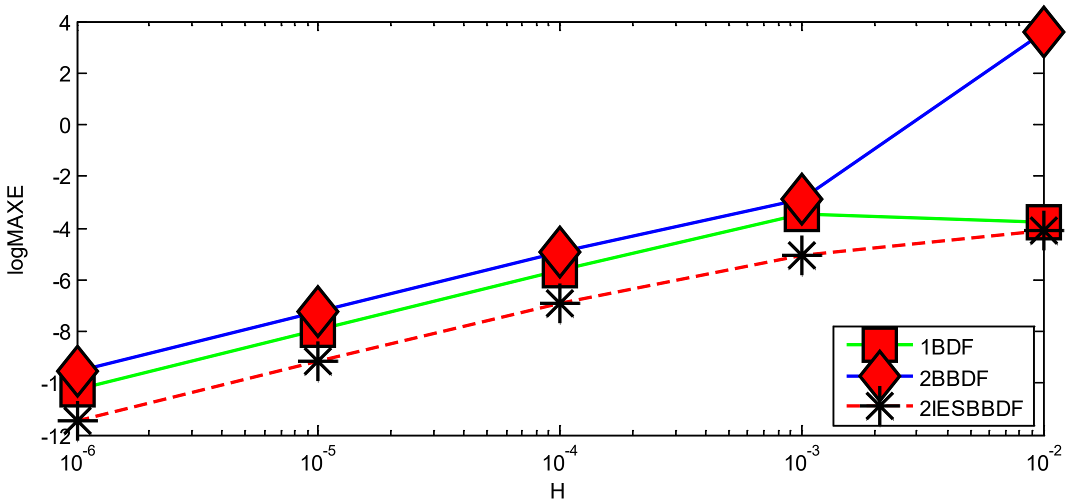

Fig. 4 *Graph of $Log_{10}MAXE$ against H for problem 3*

The charts indicate that, across nearly all examined problems, the IE2SBBDF methods exhibit smaller scaled errors compared to both the 1BDF and 2BBDF methods. This suggests an enhanced stability and accuracy in the IE2SBBDF method.

6. Conclusion

In Summary, a novel approach for tackling stiff initial value problems is presented, known as the Improved Extended 2-point Super class of Block Backward Differentiation Formula (IE2SBBDF). This method, capable of handling first-order stiff initial value problems, computes two solution values per step and boasts a fourth-order accuracy. Through stability analysis, it has been demonstrated to be nearly A-stable, zero-stable, and convergent. Extensive testing against previously reported stiff initial value problems in the literature reveals that the IE2SBBDF method surpass existing approaches such as 1BDF and 2BBDF, in terms of both accuracy and computation time. Consequently, the IE2SBBDF emerges as a viable alternative for addressing stiff initial value problems.

Acknowledgement

The authors wish to thank Department of Mathematics and Statistics, Umaru Musa Yar'adua University, Katsina for the support they received during the research.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

Study conception and design: *H.M;* ***data collection:*** *H.M, A.A;* ***analysis and interpretation of results:*** *H.M, A.A;* ***draft manuscript preparation:*** *A.A. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Buhari Alhassan, Hamza Yusuf, Musa Hamisu, & Abasi Naghmeh (2023). A New Fifth Order Variable Step Size Block Backward Differentiation Formula with Off-Step Points for the Numerical Solution of Stiff Ordinary Differential Equations. Applied Mathematics and Computational Intelligence, 12(4), 94–121.

[2] Aminikhah Hossein, & Hemmatnezhad Milad (2011). An Effective Modification of the Homotopy Perturbation Method for Stiff Systems of Ordinary Differential Equations. Applied Mathematics Letters, 24(9), 1502–1508.

[3] Abdullahi R. Yaakub & David J. Evans (2003) New L-Stable Modified Trapezoidal Methods For The Initial Value Problems, International Journal of Computer Mathematics, 80:1, 95-104, DOI:10.1080/00207160304663

[4] Musa Hamisu & Suleiman Mohamed (2014). A New Fifth Order Implicit Block Method for Solving First Order Stiff Ordinary Differential Equations. Malaysia Journal of Mathematical Sciences, 8(S), 45–59.

[5] Suleiman Mohamed Bin, Musa Hamisu, & Ismail Fudziah (2015). An Implicit 2-Point Block Extended Backward Differentiation Formula for Integration of Stiff Initial Value Problems. Malaysia Journal of Mathematical Sciences, 9(1), 33–51.

[6] Kushnir Dan, & Rokhlin Vladimir (2012). A Highly Accurate Solver for Stiff Ordinary Differential Equations. SIAM Journal on Scientific Computing, 34(3), A1296–A1315.

[7] Ibrahim Zarina Binti, Othman Kamarul Iman and Suleiman Mohamed Bin (2007): Implicit r-point block backward differentiation formula for first order stiff ODEs. Applied Mathematics and Computation, 186(1):558-565.

[8] Lee Hung-Chang, Chan Chung-Kai, Hung Ching-I (2002): A modified group-preserving scheme for solving the initial value problems of stiff ordinary differential equations. Applied mathematics and Computation. 133(2-3): 445-459.

[9] Musa Hamisu, Labaran Zainab, Ibrahim Zerina Bibi and Ibrahim Lawal Kane (2020): Extended 2-point super class of block backward differentiation formula for solving stiff initial value problems. Abacus (Journal of the Mathematical Association of Nigeria), Volume 47, Number 1 Mathematical Sciences Series.

[10] Yusuf Hamza, Hamisu Musa, & Buhari Alhassan (2024). A New Fixed Coefficient Diagonally Implicit Block Backward Differentiation Formula for Solving Stiff Initial Value Problems. UMYU Scientifica, 3(1).

[11] Musa Hamisu, Omar Zainab, Suleiman Mohamed, & Ismail Fudziah (2013). An Accurate Block Solver for Stiff Initial Value Problems. Abstract and Applied Analysis, 2013, Article ID 567451.

[12] Babangida Bature & Musa Hamisu (2016). Diagonally Implicit Super Class of Block Backward Differentiation Formula with Off-Step Points for Solving Stiff Initial Value Problems. Journal of Applied & Computational Mathematics, 5(5).

[13] Ijam Hazizah Mohd, Ibrahim Zarina Bibi, & Zawawi Iskandar Shah Mohd (2024). Stiffly Stable Diagonally Implicit Block Backward Differentiation Formula with Adaptive Step Size Strategy for Stiff Ordinary Differential Equations. MATEMATIKA, 40(1), 27–47.

[14] Suleiman Mohamed Bin, Musa Hamisu, Isma'il Fudziah, Senu Nura and Ibrahim Zarina Bibi (2014): A new superclass of block backward differentiation formulae for stiff ODEs, Asian European journal of mathematic, vol., 7:1350034.

[15] Buhari Alhassan & Hamisu Musa. (2023). Diagonally Implicit Extended 2-Point Super Class of Block Backward Differentiation Formula with Two Off-step Points for Solving First Order Stiff Initial Value Problems. Applied Mathematics and Computational Intelligence (AMCI), 12(1), 101–124.

[16] Alhassan Buhari, Musa Hamisu, Yusuf Hamza, Adamu Abdulrahman, Bello Abdullahi & Hamisu Aminu (2024). Derivation and Implementation of A-stable Diagonally Implicit Hybrid Block Method for the Numerical Integration of Stiff Ordinary Differential Equations. Matrix Science Mathematics, 8(2), 55-64.

[17] Alhassan Buhari & Musa Hamisu (2024). Super Class of Implicit Extended Backward Differentiation Formulae for the Numerical Integration of Stiff Initial Value Problems. Computational Algorithms and Numerical Dimensions, doi: 10.22105/cand.2024.487571.1157

[18] Lambert James Douglas (1973): Computational methods in ODEs, John wiley & Sons, New York.

[19] Lambert James Douglas (1991): Numerical methods for ordinary differential systems. New York: John Wiley & Sons.
[20] Rainer Alt, A-stable one-step methods with step-size control for stiff systems of ordinary differential equations, Journal of Computational and Applied Mathematics. 4 (1978), pp. 29-35.
[21] Darvishi Mohammad, Khani Farzad and Soliman Ahmed (2007): The numerical simulation for stiff systems of ordinary differential equations. Computers and Mathematics with Application 54:1055-1063.