FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 9 No. 7, July, 2025, pp 275 – 282
DOI: https://doi.org/10.33003/fjs-2025-0907-3799

A NEW 2-POINT DIAGONALLY IMPLICIT VARIABLE STEP SIZE SUPER CLASS OF BLOCK BACKWARD DIFFERENTIATION FORMULA FOR SOLVING FIRST ORDER STIFF INITIAL VALUE PROBLEMS

[1]Bala Najamuddeen, [2]Musa Hamisu and [2]Sani Abba

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

*Corresponding authors' email: balanajamuddeen@gmail.com

ABSTRACT

A new 2-point diagonally implicit variable step size super class of block backward differentiation formula (2DVSSBBDF) for solving first order stiff initial value problems (IVPs) is developed. The method is derived by introducing a lower triangular matrix in the coefficient matrix of existing 2-point variable step size superclass of block backward differentiation formula for the integration of stiff IVPs. The order of the method is 4. The stability analysis indicates that the method is both zero and A-stable. The Numerical results obtained are compared with some existing built in Matlab ODEs solvers in particular ODE15s and ODE23s and the performance of the new scheme showed an advantage in accuracy and computation time over some existing algorithms. The new method can serve as an alternative and efficient method for solving stiff IVPs.

Keywords: Diagonally implicit block method, Variable step size, Stiff, Order, Zero stability, Blocks backward differentiation formula, A–Stability

INTRODUCTION

In this paper, the general form of system of first-order stiff IVPs in Ordinary Differential Equations (ODEs) of the following form is considered:

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

where the function $f(x, y)$ is assumed to be continuously differentiable over the aforementioned interval of integration and satisfies the Lipschitz condition for the existence and uniqueness of the solution of the differential equation (1), the system (1) arises frequently in the study of fluid dynamics, control and dynamical systems, reaction kinetics, electrical circuits, combustion and so on. The system (1) is said to be stiff if it contains a very past component as well as very slow component (Dahlquist, 1978). Lambert (1991) states that, if a numerical method with a finite region of absolute stability, applied to a system with any initial condition, is forced to use a step length which is excessively small in relation to the smoothness of the exact solution in a certain interval of integration, then the system is said to be stiff in that interval. The stiffness property prevents the conventional explicit method from handling the problem efficiently, except method with A- Stability property. In solving such problems stability rather than the accuracy determines the choice of a step size. Therefore there is an increasing demand in developing implicit methods for such problems. Many numerical methods have been developed to solve (1) sequentially in (Cash, 1980; Gear, 1965; Lambert, 1991). There are other classes of methods suggested that computes a block of approximations simultaneously such as those in (Suleiman et al. (2014); Majid et al. (2006); Musa et al. (2012); Musa and Bala (2019); Musa and Unwala (2019); Alhassan and Musa (2023), Yusuf et al. (2024)). This paper presents the derivation of diagonally implicit form of the method in Suleiman et al. (2013), by introducing a lower triangular matrix in the method for the purpose of reducing the number of function evaluations in the computation which would obviously save the computation time.

METHODS AND MATERIALS

Derivation of the Method

Consider a new variable step size block backward differentiation formula for solving stiff initial value problems developed by Suleiman *et al* (2013):

$$\sum_{j=o}^{4} \alpha_{j,i,r} y_{n+j-2} = h\beta_{k,i,r}(f_{n+k} - \rho f_{n+k-1}) \qquad k = i = 1,2 \qquad (2)$$

where, $\beta_{k-1,i,r} = \rho\beta_{k,i,r}$. $\rho \,\epsilon (-1,1)$. Formula (2) is known for the numerical integration of stiff initial value problems and by chosen ρ within the interval$(-1,1)$, the formula is A-stable. The method is fully implicit and produces 2 point per each step.

In this paper, we introduce a lower triangular matrix in the previous method (2), there by defining a new 2- point diagonally implicit variable step size super class of block backward differentiation formula as follows:

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

$k = i = 1$, represents the first point formula while $k = i = 2$, represents the second point formula. The formula (3) is derived using Taylor series expansion as follows:

Definition 1

The Linear operator L_i associated with first point of a new 2-point diagonally implicit variable step size super class of block BDF method is defined as:

$$L_i[y(x_n), h]: \alpha_{0,i} y(x_n - 2rh) + \alpha_{1,i} y(x_n - rh) + \alpha_{2,i} y(x_n) + \alpha_{3,i} y(x_n + h) - h\beta_{k,i} f(x_n + kh) + h\rho\beta_{k,i} f(x_n + (k-1)h) = 0, k = i = 1 \qquad (4)$$

The linear operator (4) becomes

$$L_1[y(x_n), h]: \alpha_{0,1} y(x_n - 2rh) + \alpha_{1,1} y(x_n - rh) + \alpha_{2,1} y(x_n) + \alpha_{3,1} y(x_n + h) - h\beta_{1,1} y'(x_n + h) + h\rho\beta_{1,1} y'(x_n) = 0, \qquad (5)$$

Expanding the functions in equation (5) as Taylor series and collecting like terms give:$C_{0,1} y(x_n) + C_{1,1} hy'(x_n) + C_{2,1} h^2 y''(x_n) + C_{3,1} h^3 y'''(x_n) \ldots = 0. \qquad (6)$

where

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

In deriving the first point y_{n+1} the coefficient $\alpha_{3,1}$ is normalized to 1. Solving the system of equations (7) for the values of $\alpha_{j,i}$ and $\beta_{j,i}$ gives

$$\alpha_{0,1} = \frac{1}{2}\frac{r^2\rho+r^2+r\rho+2r+1}{r^2(2r^2\rho-2r^2-6r-3)},$$

$$\alpha_{1,1} = -\frac{4r^2\rho+4r^2+2r\rho+4r+1}{r^2(2r^2\rho-2r^2-6r-3)},$$

$$\alpha_{2,1} = -\frac{1}{2}\frac{4r^4\rho-4r^4-12r^3-7r^2\rho-13r^2-3r\rho-6r-1}{r^2(2r^2\rho-2r^2-6r-3)},$$

$$\alpha_{3,1} = 1,\ \beta_{1,1} = -\frac{2r^2+3r+1}{2r^2\rho-2r^2-6r-3}$$

Substituting these values in equation (5), we obtain

$$y_{n+1} = -\frac{1}{2}\frac{r^2\rho+r^2+r\rho+2r+1}{r^2(2r^2\rho-2r^2-6r-3)}y_{n-2} + \frac{4r^2\rho+4r^2+2r\rho+4r+1}{r^2(2r^2\rho-2r^2-6r-3)}y_{n-1} + \frac{1}{2}\frac{4r^4\rho-4r^4-12r^3-7r^2\rho-13r^2-3r\rho-6r-1}{r^2(2r^2\rho-2r^2-6r-3)}y_n - \frac{2r^2+3r+1}{2r^2\rho-2r^2-6r-3}hf_{n+1} + \frac{2r^2+3r+1}{2r^2\rho-2r^2-6r-3}h\rho f_n \quad (8)$$

Second Point: $k = i = 2$.

Definition 2

Define the Linear operator L_i associated with the second point of a new 2-point diagonally implicit variable step size super class of block BDF method as:

$L_i[y(x_n), h]$: $\alpha_{0,i}y(x_n - 2rh) + \alpha_{1,i}y(x_n - rh) + \alpha_{2,i}y(x_n) + \alpha_{3,i}y(x_n + h) + \alpha_{4,i}y(x_n + 2h) - h\beta_{k,i}f(x_n + kh) + h\rho\beta_{k,i}f(x_n + (k-1)h) = 0,\ k = i = 2$ (9)

To derive the second point y_{n+2}. The linear operator (9) becomes

$$L_2[y(x_n), h] = \alpha_{0,2}y(x_n - 2rh) + \alpha_{1,2}y(x_n - rh) + \alpha_{2,2}y(x_n) + \alpha_{3,2}y(x_n + h) + \alpha_{4,2}y(x_n + 2h) - h\beta_{2,2}y'(x_n + h) + h\rho\beta_{2,2}y'(x_n + h) = 0. \quad (10)$$

Expanding (10) as a Taylor's series about x_n and collecting the like terms give

$$C_{0,2}y(x_n) + C_{1,2}hy'(x_n) + C_{2,2}h^2y''(x_n) + C_{3,2}h^3y'''(x_n) + \cdots = 0 \quad (11)$$

where

$$\left.\begin{aligned} C_{0,2} &= \alpha_{0,2} + \alpha_{1,2} + \alpha_{2,2} + \alpha_{3,2} + \alpha_{4,2} = 0 \\ C_{1,2} &= -2r\alpha_{0,2} - r\alpha_{1,2} + \alpha_{3,2} + 2\alpha_{4,2} - \beta_{2,2}(1-\rho) = 0 \\ C_{2,2} &= 2r^2\alpha_{0,2} + \tfrac{1}{2}r^2\alpha_{1,2} + \tfrac{1}{2}\alpha_{3,2} + 2\alpha_{4,2} - \beta_{2,2}(2-\rho) = 0 \\ C_{3,2} &= -\tfrac{4}{3}r^3\alpha_{0,2} - \tfrac{1}{6}r^3\alpha_{1,2} + \tfrac{1}{6}\alpha_{3,2} + \tfrac{4}{3}\alpha_{4,2} - \beta_{2,2}\left(2 - \tfrac{1}{2}\rho\right) = 0 \\ C_{4,2} &= \tfrac{2}{3}r^4\alpha_{0,2} + \tfrac{1}{24}r^4\alpha_{1,2} + \tfrac{1}{24}\alpha_{3,2} + \tfrac{2}{3}\alpha_{4,2} - \beta_{2,2}\left(\tfrac{4}{3} - \tfrac{1}{6}\rho\right) = 0 \end{aligned}\right\}. \quad (12)$$

In deriving the second point y_{n+2} the coefficient $\alpha_{4,2}$ is normalized to 1. Solving the system of equations (12) for the values of $\alpha_{j,i}$ and $\beta_{j,i}$ gives

$$\alpha_{0,2} = -\frac{r^2\rho + 2r^2 + 3r\rho + 8r + 2\rho + 8}{r^2(2r+1)(2r^2\rho - 6r^2 + 3r\rho - 24r + \rho - 20)},$$

$$\alpha_{1,2} = -\frac{2r^4\rho + 2r^4 + 9r^3\rho + 12r^3 + 14r^2\rho + 26r^2 + 9r\rho + 24r + 2\rho + 8}{r^2(2r^2\rho - 6r^2 + 3r\rho - 24r + \rho - 20)},$$

$$\alpha_{2,2} = \frac{4(2r\rho + 4r + \rho + 4)}{r^2(2r^2\rho - 6r^2 + 3r\rho - 24r + \rho - 20)}, \quad \alpha_{3,2} = 1,$$

$$\alpha_{4,2} = \frac{4(4r^3\rho + 3r^2\rho + 20r^2\rho + 8r\rho + 32r + 4\rho + 16)}{(2r+1)(2r^2\rho - 6r^2 + 3r\rho - 24r + \rho - 20)},$$

$$\beta_{2,2} = -\frac{4(r^2 + 3r + 2)}{(2r^2\rho - 6r^2 + 3r\rho - 24r + \rho - 20)}$$

Substituting these values in equation (10), we obtain

$$y_{n+2} = \frac{r^2\rho+2r^2+3r\rho+8r+2\rho+8}{r^2(2r+1)(2r^2\rho-6r^2+3r\rho-24r+\rho-20)}y_{n-2} + \frac{2r^4\rho+2r^4+9r^3\rho+12r^3+14r^2\rho+26r^2+9r\rho+24r+2\rho+8}{r^2(2r^2\rho-6r^2+3r\rho-24r+\rho-20)}y_{n-1} - \frac{4(2r\rho+4r+\rho+4)}{r^2(2r^2\rho-6r^2+3r\rho-24r+\rho-20)}y_n + \frac{4(4r^3\rho+3r^2\rho+20r^2\rho+8r\rho+32r+4\rho+16)}{(2r+1)(2r^2\rho-6r^2+3r\rho-24r+\rho-20)}y_{n+1} - \frac{4(r^2+3r+2)}{(2r^2\rho-6r^2+3r\rho-24r+\rho-20)}hf_{n+2} + \frac{4(r^2+3r+2)}{(2r^2\rho-6r^2+3r\rho-24r+\rho-20)}h\rho f_{n+1} \quad (13)$$

For stability reasons, the value of the free parameter ρ is chosen within the interval $(-1,1)$, the value of ρ is taken to be $-\frac{3}{4}$ and by substituting $r = 1, r = 2$ and $r = \frac{5}{8}$ in equation (8) and (13) gives the coefficients of the method as given below;

For $r = 1$

$$\left.\begin{aligned} y_{n+1} &= \tfrac{1}{10}y_{n-2} - \tfrac{9}{25}y_{n-1} + \tfrac{63}{50}y_n + \tfrac{12}{25}hf_{n+1} + \tfrac{9}{25}hf_n \\ y_{n+2} &= -\tfrac{9}{109}y_{n-2} + \tfrac{46}{109}y_{n-1} - \tfrac{90}{109}y_n + \tfrac{162}{109}y_{n+1} + \tfrac{48}{109}hf_{n+2} + \tfrac{36}{109}hf_{n+1} \end{aligned}\right\} \quad (14)$$

For $r = 2$

$$\left.\begin{aligned} y_{n+1} &= \tfrac{9}{464}y_{n-2} - \tfrac{5}{58}y_{n-1} + \tfrac{495}{464}y_n + \tfrac{15}{29}hf_{n+1} + \tfrac{45}{116}hf_n \\ y_{n+2} &= -\tfrac{23}{2065}y_{n-2} + \tfrac{33}{413}y_{n-1} - \tfrac{153}{413}y_n + \tfrac{384}{295}y_{n+1} + \tfrac{192}{413}hf_{n+2} + \tfrac{144}{413}hf_{n+1} \end{aligned}\right\} \quad (15)$$

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

$$\left.\begin{aligned} y_{n+1} &= \tfrac{7696}{25975}y_{n-2} - \tfrac{24192}{25975}y_{n-1} + \tfrac{42471}{25975}y_n + \tfrac{468}{1039}hf_{n+1} + \tfrac{351}{1039}hf_n \\ y_{n+2} &= -\tfrac{5504}{18325}y_{n-2} + \tfrac{22528}{18325}y_{n-1} - \tfrac{28899}{18325}y_n + \tfrac{1208}{733}y_{n+1} + \tfrac{312}{733}hf_{n+2} + \tfrac{234}{733}hf_{n+1} \end{aligned}\right\} \quad (16)$$

The order of the method is 4 with the following error constants corresponding to the three different values of r (that is $r = 1, 2\ and\ \frac{5}{8}$) respectively

$$C_5 = \begin{bmatrix} -\frac{129}{500} \\ -\frac{39}{545} \end{bmatrix} \quad C_5 = \begin{bmatrix} \frac{7}{65} \\ -\frac{4}{743} \end{bmatrix} \text{ and } C_5 = \begin{bmatrix} \frac{1}{23} \\ -\frac{37}{914} \end{bmatrix}$$

Stability Analysis of the Method

The stability properties of the methods (14), (15), and (16) are investigated, focusing on both zero and A-stability. To be considered practically useful for addressing stiff problems, a method should ideally beside been zero stable, exhibit at least almost A-stability property.

The linear stability properties of the methods are determined through the application of the standard linear test first order ordinary differential equation

$$y' = \lambda y,\ \ \lambda < 0 \quad (17)$$

To investigate zero stability of method (14) when $r = 1$, then the formula (14) is represented in matrix form as:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{10} \\ 0 & -\frac{9}{109} \end{bmatrix}\begin{bmatrix} y_{n-3} \\ y_{n-2} \end{bmatrix} + \begin{bmatrix} -\frac{9}{25} & \frac{63}{50} \\ \frac{46}{109} & -\frac{90}{109} \end{bmatrix}\begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{162}{109} & 0 \end{bmatrix}\begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix} + h\begin{bmatrix} 0 & \frac{9}{25} \\ 0 & 0 \end{bmatrix}\begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix} + h\begin{bmatrix} \frac{12}{25} & 0 \\ \frac{36}{109} & \frac{48}{109} \end{bmatrix}\begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix} \quad (18)$$

This is equivalent to the following matrix equation:

$$\begin{bmatrix}1&0\\-\frac{162}{109}&1\end{bmatrix}\begin{bmatrix}y_{n+1}\\y_{n+2}\end{bmatrix}=\begin{bmatrix}0&\frac{1}{10}\\0&-\frac{9}{109}\end{bmatrix}\begin{bmatrix}y_{n-3}\\y_{n-2}\end{bmatrix}+\begin{bmatrix}-\frac{9}{25}&\frac{63}{50}\\\frac{46}{109}&-\frac{90}{109}\end{bmatrix}\begin{bmatrix}y_{n-1}\\y_n\end{bmatrix}+h\begin{bmatrix}0&\frac{9}{25}\\0&0\end{bmatrix}\begin{bmatrix}f_{n-1}\\f_n\end{bmatrix}+h\begin{bmatrix}\frac{12}{25}&0\\\frac{36}{109}&\frac{48}{109}\end{bmatrix}\begin{bmatrix}f_{n+1}\\f_{n+2}\end{bmatrix} \quad (19)$$

It can be noted that the coefficient matrix on the left hand side of (19) is a lower triangular matrix, hence qualifying the method to be called diagonally implicit. Putting the linear test equation (17) in (19) and simplifying, we have

$$\begin{bmatrix}\left(1-\frac{12}{25}h\lambda\right)&0\\\left(-\frac{162}{109}-\frac{36}{109}h\lambda\right)&\left(1-\frac{48}{109}h\lambda\right)\end{bmatrix}\begin{bmatrix}y_{n+1}\\y_{n+2}\end{bmatrix}=\begin{bmatrix}0&\frac{1}{10}\\0&-\frac{9}{109}\end{bmatrix}\begin{bmatrix}y_{n-3}\\y_{n-2}\end{bmatrix}+\begin{bmatrix}-\frac{9}{25}&\left(\frac{63}{50}+\frac{9}{25}h\lambda\right)\\\frac{46}{109}&-\frac{90}{109}\end{bmatrix}\begin{bmatrix}y_{n-1}\\y_n\end{bmatrix} \quad (20)$$

Letting $h\lambda=\bar{h}$ into (20) gives

$$\begin{bmatrix}\left(1-\frac{12}{25}\bar{h}\right)&0\\\left(-\frac{162}{109}-\frac{36}{109}\bar{h}\right)&\left(1-\frac{48}{109}\bar{h}\right)\end{bmatrix}\begin{bmatrix}y_{n+1}\\y_{n+2}\end{bmatrix}=\begin{bmatrix}0&\frac{1}{10}\\0&-\frac{9}{109}\end{bmatrix}\begin{bmatrix}y_{n-3}\\y_{n-2}\end{bmatrix}+\begin{bmatrix}-\frac{9}{25}&\left(\frac{63}{50}+\frac{9}{25}\bar{h}\right)\\\frac{46}{109}&-\frac{90}{109}\end{bmatrix}\begin{bmatrix}y_{n-1}\\y_n\end{bmatrix} \quad (21)$$

This equation is equivalent to

$$AY_m=BY_{m-1}+CY_{m-2},\ n=2m, \quad (22)$$

where,

$$A=\begin{bmatrix}\left(1-\frac{12}{25}\bar{h}\right)&0\\\left(-\frac{162}{109}-\frac{36}{109}\bar{h}\right)&\left(1-\frac{48}{109}\bar{h}\right)\end{bmatrix},$$

$$B=\begin{bmatrix}-\frac{9}{25}&\left(\frac{63}{50}+\frac{9}{25}\bar{h}\right)\\\frac{46}{109}&-\frac{90}{109}\end{bmatrix},\ C=\begin{bmatrix}0&\frac{1}{10}\\0&-\frac{9}{109}\end{bmatrix},$$

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

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

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

The stability polynomial of the method is obtained by computing the determinant of $At^2+Bt+C=0$

$$\pi_1(t,\bar{h})=-\frac{819}{2725}t^2-\frac{34}{2725}t-\frac{1872}{2725}t^3+t^4-\frac{612}{2725}\bar{h}t^2-\frac{324}{2725}\bar{h}^2t^3+\frac{576}{2725}\bar{h}^2t^4-\frac{2508}{2725}\bar{h}t^4-\frac{4104}{2725}\bar{h}t^3=0 \quad (23)$$

For zero stability, we set $\bar{h}=0$ in equation (23) to obtain the first characteristics polynomial as

$$\pi_1(t,0)=t^4-\frac{1872}{2725}t^3-\frac{819}{2725}t^2-\frac{34}{2725}t=0 \quad (24)$$

Solving (24) for t, leads to the following roots:

$t=0, t=1, t=-0.2661472113, t=-0.0468803117$

These values of t indicate that the method is zero stable.

Adopting similar procedure for the values of r = 2, then the characteristics polynomial is obtained as

$$\pi_2(t,\bar{h})=-\frac{12919}{191632}t^2-\frac{113}{191632}t-\frac{22325}{23954}t^3+t^4-\frac{2085}{47908}\bar{h}t^2-\frac{1620}{11977}\bar{h}^2t^3+\frac{2880}{11977}\bar{h}^2t^4-\frac{11763}{11977}\bar{h}t^4-\frac{13278}{11977}\bar{h}t^3=0 \quad (25)$$

For zero stability, we set $\bar{h}=0$ in equation (25), resulting in the first characteristics polynomial as follows:

$$\pi_2(t,0)=t^4-\frac{22325}{23954}t^3-\frac{12919}{191632}t^2-\frac{113}{191632}t=0 \quad (26)$$

Solving equation (26) for t, leads to the following roots:

$t=0, t=1, t=-0.05780413850, t=-0.01020120508$

The values of t indicates that the method is zero-stable.

Using the same procedure for the value of $r=5/8$, then the characteristic polynomial is obtained as

$$\pi_3(t,\bar{h})=-\frac{69422976}{95198375}t^2-\frac{8044544}{95198375}t-\frac{3546171}{19039675}t^3+t^4-\frac{12284064}{19039675}\bar{h}t^2-\frac{82134}{761587}\bar{h}^2t^3+\frac{146016}{761587}\bar{h}^2t^4-\frac{667212}{761587}\bar{h}t^4-\frac{1664442}{761587}\bar{h}t^3=0 \quad (27)$$

To show that method is zero stable, we set $\bar{h}=0$ in equation (27), resulting in the first characteristics polynomial as follows:

$$\pi_3(t,\bar{h})=t^4-\frac{3546171}{19039675}t^3-\frac{69422976}{95198375}t^2-\frac{8044544}{95198375}t=0 \quad (28)$$

Solving equation (28) for t yields the following roots

$t=0, t=1, t=-0.6915558132, t=-0.1221925308$

Hence, the DVSSBBDF4 when $r=5/8$ is zero stable.

The stability region of the stability polynomial for the three different values of the step size ratio r are given in the figures below.

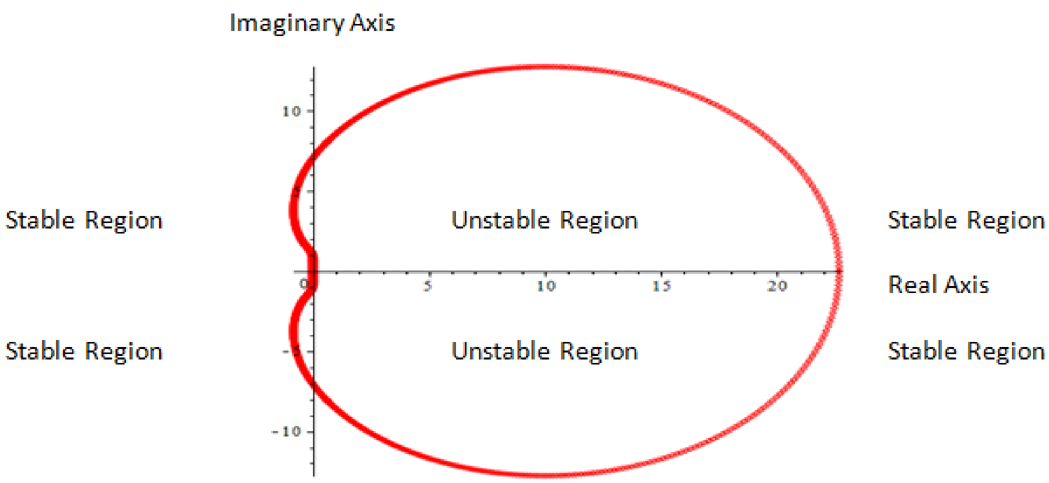

Figure 1: Stability Region for 2DVSSBBDF method when $r=1$

The stability covers almost the entire negative half plane, which shows that the method is almost A-stable.

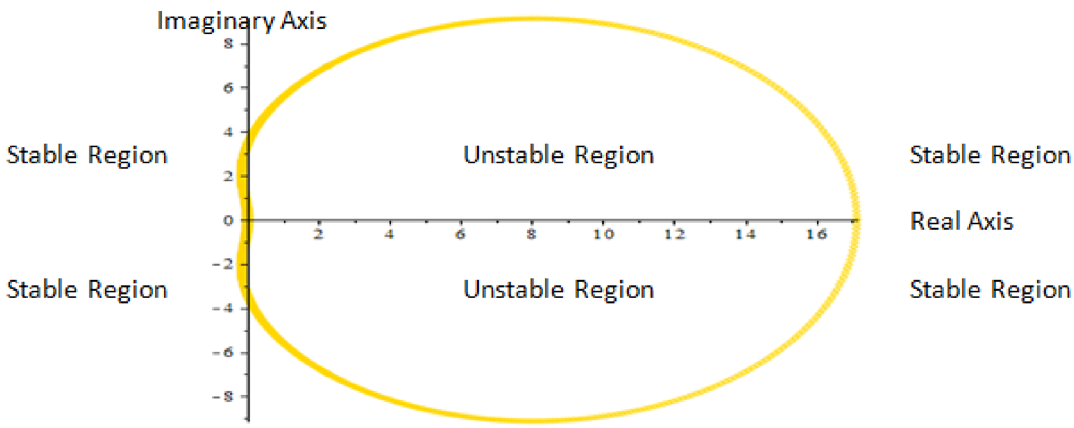

Figure 2: Stability Region for 2-DVSSBBDF method when $r = 2$

The stability region covers the entire negative half plane and hence the method is A-stable.

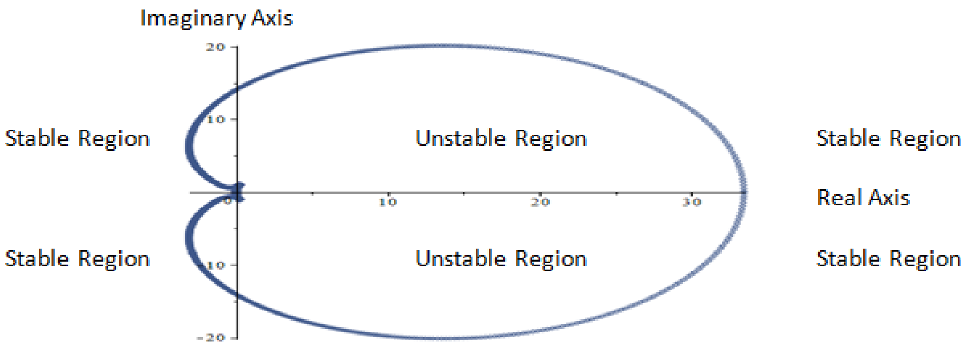

Figure 3: Stability Region for 2DVSSBBDF4 method when $r = 5/8$

The region of absolute stability is the region outside the circle and it indicates that method (16) is almost A-stable since the stability region covers almost the entire negative half plane. One of the desirable properties for the numerical solution of stiff initial value problems (IVPs) is A-stability. Hence, the methods developed are suitable for the numerical integration of stiff initial value problems.

Implementation of the Method

Newton's iteration is employed to implement the method. The procedure is described as follows. The formulae (14)–(16) can be represented in the following form:

$$\left.\begin{aligned} y_{n+1} &= \theta_1 y_{n+2} + \alpha_1 h f_n + \alpha_2 h f_{n+1} + \tau_1 \\ y_{n+2} &= \theta_2 y_{n+1} + \alpha_3 h f_{n+1} + \alpha_4 h f_{n+2} + \tau_2 \end{aligned}\right\} \quad (29)$$

where τ_1 and τ_2 are the back values.

Equation (29) can be rewritten as

$$\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}y_{n+1}\\ y_{n+2}\end{bmatrix} = \begin{bmatrix}0 & \theta_1\\ \theta_2 & 0\end{bmatrix}\begin{bmatrix}y_{n+1}\\ y_{n+2}\end{bmatrix} + h\begin{bmatrix}0 & \alpha_1\\ 0 & 0\end{bmatrix}\begin{bmatrix}f_{n-1}\\ f_n\end{bmatrix} + h\begin{bmatrix}\alpha_2 & 0\\ \alpha_3 & \alpha_4\end{bmatrix}\begin{bmatrix}f_{n+1}\\ f_{n+2}\end{bmatrix} + \begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} \quad (30)$$

which can be represented as

$$(I - A)Y = h(B_1F_1 + B_2F_2) + \Psi \quad (31)$$

where

$$I = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}, A = \begin{bmatrix}0 & \theta_1\\ \theta_2 & 0\end{bmatrix}, B_1 = \begin{bmatrix}0 & \alpha_1\\ 0 & 0\end{bmatrix},$$

$$B_2 = \begin{bmatrix}\alpha_2 & 0\\ \alpha_3 & \alpha_4\end{bmatrix}, Y = \begin{bmatrix}y_{n+1}\\ y_{n+2}\end{bmatrix}, F_1 = \begin{bmatrix}f_{n-1}\\ f_n\end{bmatrix}, F_2 = \begin{bmatrix}f_{n+1}\\ f_{n+2}\end{bmatrix}, \text{and}$$

$$\psi = \begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix}$$

Let

$$F = (I - A)Y - h(B_1F_1 + B_2F_2) - \Psi = 0 \quad (32)$$

Newton's iteration for the 2DVSSBBDF method takes the form

$$Y_{n+1,n+2}^{(i+1)} - Y_{n+1,n+2}^{(i)} = -\left(F'_j\left(y_{n+1,n+2}^{(i)}\right)\right)^{-1}\left(F_j\left(y_{n+1,n+2}^{(i)}\right)\right) \quad (33)$$

Equation (33) is equivalent to

$$Y_{n+1,n+2}^{(i+1)} - Y_{n+1,n+2}^{(i)} = -\left((I - A) - hB_1\frac{\partial F_1}{\partial Y}\left(y_{n+1,n+2}^{(i)}\right) - hB_2\frac{\partial F_2}{\partial Y}\left(y_{n+1,n+2}^{(i)}\right)\right)^{-1} \times \left((I - A)(y_{n+1,n+2}^{(i)}) - hB_1F_1 - hB_2F_2 - \Psi\right) \quad (34)$$

For comparison, the maximum error is computed from the algorithm developed. Let y_i and $y(x_i)$ be the computed and exact solution of equation (1) respectively. The absolute error is defined by

$$(error_i)_t = |(y_i)_t - (y(x_i))_t| \quad (35)$$

The maximum error is defined by:

$$MAXE = \underbrace{max}_{1\le i\le T}\left(\underbrace{max(error_i)_t}_{1\le i\le N}\right), \quad (36)$$

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

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

$$E_{1,2}^{(i+1)} = Y_{n+1,n+2}^{(i+1)} - Y_{n+1,n+2}^{(i)} \quad (37)$$

The equation (34) can be written as

$$E_{1,2}^{(i+1)} = \bar{A}^{-1}\bar{B} \qquad (38)$$

which is equivalent to

$$\bar{A}E_{1,2}^{(i+1)} = \bar{B} \qquad (39)$$

where

$$\bar{A} = \left((I-A) - hB_1\frac{\partial F_1}{\partial Y}\left(y_{n+1,n+2}^{(i)}\right) - hB_2\frac{\partial F_2}{\partial Y}\left(y_{n+1,n+2}^{(i)}\right)\right)^{-1}$$

And

$$\bar{B} = -\left((I-A)\left(y_{n+1,n+2}^{(i)}\right) - hB_1F_1 - hB_2F_2 - \Psi\right)$$

Newton's iteration would therefore be used to solve the system (39). For the different value of r.

$$\bar{A}\begin{pmatrix} 1-\alpha_2 h\frac{\partial f_{n+1}}{\partial y_{n+1}} & -\theta_1 \\ -\theta_2-\theta_3 h\frac{\partial f_{n+1}}{\partial y_{n+1}} & 1-\alpha_4 h\frac{\partial f_{n+2}}{\partial y_{n+2}} \end{pmatrix}$$

$$\bar{B}\begin{pmatrix} -y_{n+1}^i + \theta_1 y_{n+2}^i + \alpha_1 hf_n^i + \alpha_2 hf_{n+2}^i + \Psi \\ -y_{n+2}^i + \theta_2 y_{n+1}^i + \alpha_3 hf_{n+1}^i + \alpha_4 hf_{n+2}^i + \Psi \end{pmatrix}$$

When $r = 1$

$$\bar{A} = \begin{pmatrix} 1-\frac{12}{25}h\frac{\partial f_{n+1}}{\partial y_{n+1}} & 0 \\ -\frac{162}{109}-\frac{36}{109}h\frac{\partial f_{n+1}}{\partial y_{n+1}} & 1-\frac{48}{109}h\frac{\partial f_{n+2}}{\partial y_{n+2}} \end{pmatrix}$$

$$\bar{B} = \begin{pmatrix} -y_{n+1}^i + \frac{9}{25}hf_n^i + \frac{12}{25}hf_{n+1}^i + \frac{1}{10}y_{n-2} - \frac{9}{25}y_{n-1} + \frac{63}{25}y_n \\ -y_{n+2}^i + \frac{162}{109}y_{n+1}^i + \frac{36}{109}hf_{n+1}^i + \frac{48}{109}hf_{n+2}^i - \frac{9}{109}y_{n-2} + \frac{46}{109}y_{n-1} - \frac{90}{109}y_n \end{pmatrix}$$

When $r = 2$

$$\bar{A} = \begin{pmatrix} 1-\frac{15}{29}h\frac{\partial f_{n+1}}{\partial y_{n+1}} & 0 \\ -\frac{384}{295}-\frac{144}{413}h\frac{\partial f_{n+1}}{\partial y_{n+1}} & 1-\frac{192}{413}h\frac{\partial f_{n+2}}{\partial y_{n+2}} \end{pmatrix}$$

$$\bar{B} = \begin{pmatrix} -y_{n+1}^i + \frac{45}{116}hf_n^i + \frac{15}{29}hf_{n+1}^i + \frac{9}{464}y_{n-2} - \frac{5}{58}y_{n-1} + \frac{495}{464}y_n \\ -y_{n+2}^i + \frac{384}{295}y_{n+1}^i + \frac{144}{413}hf_{n+1}^i + \frac{192}{413}hf_{n+2}^i - \frac{23}{2065}y_{n-2} + \frac{33}{413}y_{n-1} - \frac{153}{413}y_n \end{pmatrix}$$

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

$$\bar{A} = \begin{pmatrix} 1-\frac{468}{1039}h\frac{\partial f_{n+1}}{\partial y_{n+1}} & 0 \\ -\frac{1208}{733}-\frac{234}{733}h\frac{\partial f_{n+1}}{\partial y_{n+1}} & 1-\frac{312}{733}h\frac{\partial f_{n+2}}{\partial y_{n+2}} \end{pmatrix}$$

$$\bar{B} = \begin{pmatrix} -y_{n+1}^i + \frac{351}{1039}hf_n^i + \frac{468}{1039}hf_{n+1} + \frac{7696}{25975}y_{n-2} - \frac{24192}{25975}y_{n-1} + \frac{42471}{25975}y_n \\ -y_{n+2}^i + \frac{1208}{733}y_{n+1}^i + \frac{234}{733}hf_{n+1}^i + \frac{312}{733}hf_{n+2}^i - \frac{5504}{18325}y_{n-2} + \frac{22528}{18325}y_{n-1} - \frac{28899}{18325}y_n \end{pmatrix}$$

Step Size Selection

To enhance the performance and efficiency of the BBDF algorithm, the algorithm in this paper is designed to have the capacity of varying the step size. The importance of choosing the appropriate step size is to achieve reduction in computation time and number of iterations. Three techniques are engaged in controlling the step size, starting with initial step size determination and local truncation error computation during the integration process. The local truncation error (LTE) is computed as follows:

$$LTE = \left|y_{n+2}^{(p+1)} - y_{n+2}^{(p)}\right| \quad k = 2,3,4.$$

Where $y_{n+2}^{(p+1)}$ refers to the formula of order *p+1* and $y_{n+2}^{(p)}$ refers to the formula of order *p*.

If the local truncation error (LTE) is less than or equal to a specified error tolerance (TOL) (i.e. LTE ≤ TOL), the step size (from previous block) is maintained as constant (equivalent to the formula when *r* = 1) and the following is computed:

$$h_{new} = c \times h_{old} \times \left(\frac{TOL}{LTE_p}\right)^{\frac{1}{P}}$$

Where c is the safety factor, which is considered as 0.5, p is the order of the method while h_{old} and h_{new} are the step size from the previous and current block respectively.

If $h_{new} > (1.6 \times h_{old})$, then the step size *h* becomes $h = 1.6 \times h_{old}$ (equivalent to the formula when $r = \frac{5}{8}$).

Otherwise, if LTE > TOL, the step size is halved (equivalent to the formula when r = *2*).

Test Problems

To test the performance of the new method developed, the following stiff IVPs are considered.

Problem 1:

$y' = -20y + 24, \quad y(0) = 0, \qquad 0 \le x \le 10$

Exact solution:

$y_1(x) = \frac{6}{5} - \left(\frac{6}{5}\right)e^{-20x},$

Eigen Values: $\lambda = -20$

Source: Yatim et al (2011).

Problem 2:

$y'_1 = 998y_1 + 1998y_2, \qquad y_1(0) = 1, \;\; 0 \le x \le 20,$

$y'_2 = -999y_1 - 1999y_2\,, \qquad y_2(0) = 0.$

Exact solution:

$y_1(x) = 2e^{-x} - e^{-1000x},$

$y_2(x) = -e^{-2x} + e^{-1000x}.$

Eigen Values: $\lambda_1 = -1$ and $\lambda_2 = -1000$

Source: Yatim et al (2011)

Problem 3:

$y'_1 = -20y_1 - 0.25y_2 - 19.75y_3, \qquad y_1(0) = 1, \;\; 0 \le x \le 10,$

$y'_2 = 20y_1 - 20.25y_2 + 0.25y_3, \qquad y_2(0) = 0,$

$y'_3 = 20y_1 - 19.75y_2 - 0.25y_3,$

$y_3(0) = -1.$

Exact solution:

$y_1(x) = 0.5\left(e^{-0.5x} + e^{-20x}(\cos 20x + \sin 20x)\right),$

$y_2(x) = 0.5\left(e^{-0.5x} - e^{-20x}(\cos 20x - \sin 20x)\right),$

$y_2(x) = -0.5\left(e^{-0.5x} + e^{-20x}(\cos 20x - \sin 20x)\right).$

Eigen Values: $\lambda_1 = -0.5, \; \lambda_2 = -20 + 20i,$ and $\lambda_3 = -20 - 20i.$

Source: Yatim et al (2011).

Numerical Results and Discussion

Given below are tables of results for the test problems presented in the previous section. All the problems are solved with the MATLAB built in solvers namely, ode15s and ode23s for comparison purpose and then the new developed method i.e. A new 2-point diagonally implicit variable step size super class of block backward differentiation formula. The number of steps taken to complete the integration, the maximum error and the time taken using each method are given and compared. Graph of $Log_{10}(MAXE)$ versus TOL are also plotted for each problem.

Abbreviations used in the tables are:

TOL: Tolerance Value Used
METHOD: The Methods Used
TS: Total number of integration steps
MAXE: Maximum Error
CPU TIME: C omputation Time in Seconds
2DVSSBBDF: A new 2-point diagonally implicit variable step size super class of block

Backward Differentiation Formula

ODE15S: A variable order (1-5) solver based on the numerical differentiation formulae (NDF)
OD23S:A modified Rosenbrock of order two formula

Table 1: Numerical Result for Problem 1

TOL	METHOD	TS	MAXE	CPU TIME
10^{-2}	ODE15S	29	8.70000E-003	3.13000E-002
	ODE23S	19	4.80000E-003	2.81300E-001
	2DVSSBBDF	46	1.76164E-004	2.27100E-004
10^{-4}	ODE15S	61	1.70460E-004	3.13000E-002
	ODE23S	43	2.74000E-004	1.56000E-002
	2DVSSBBDF	60	4.36547E-005	4.06120E-004
10^{-6}	ODE15S	96	2.71750E-006	4.69000E-002
	ODE23S	148	1.33090E-005	3.13000E-002
	2DVSSBBDF	90	1.67330E-006	7.32760E-004

Table 2: Numerical Result for Problem 2

TOL	METHOD	TS	MAXE	CPU TIME
10^{-2}	ODE15S	38	1.76000E-002	4.69000E-002
	ODE23S	23	7.30000E-003	1.40600E-002
	2DVSSBBDF	48	2.92585E-004	1.54310E-004
10^{-4}	ODE15S	90	1.86590E-004	1.56000E-002
	ODE23S	68	3.68370E-004	6.25000E-002
	2DVSSBBDF	61	4.13979E-005	7.12730E-004
10^{-6}	ODE15S	162	3.95690E-006	6.25000E-002
	ODE23S	288	1.70390E-005	7.81000E-002
	2DVSSBBDF	79	2.03559E-006	4.53210E-003

Table 3: Numerical Result for Problem 3

TOL	METHOD	TS	MAXE	CPU TIME
10^{-2}	ODE15S	34	1.09000E-002	3.13000E-002
	ODE23S	21	6.40000E-003	3.13000E-002
	2DVSSBBDF	43	4.30894E-004	1.22940E-004
10^{-4}	ODE15S	71	2.02740E-004	4.69000E-002
	ODE23S	55	3.53750E-004	1.56000E-002
	2DVSSBBDF	59	5.05315E-005	5.63120E-003
10^{-6}	ODE15S	140	4.00590E-006	4.69000E-002
	ODE23S	233	1.70230E-005	1.25000E-001
	2DVSSBBDF	74	2.64856E-006	8.93560E-003

In order to give a visual impact of the performance of the method, the graphs of $Log_{10}(MAXE)$ against TOL for the problems tested are plotted. Given below are the graphs of scaled maximum error for the different problems tested.

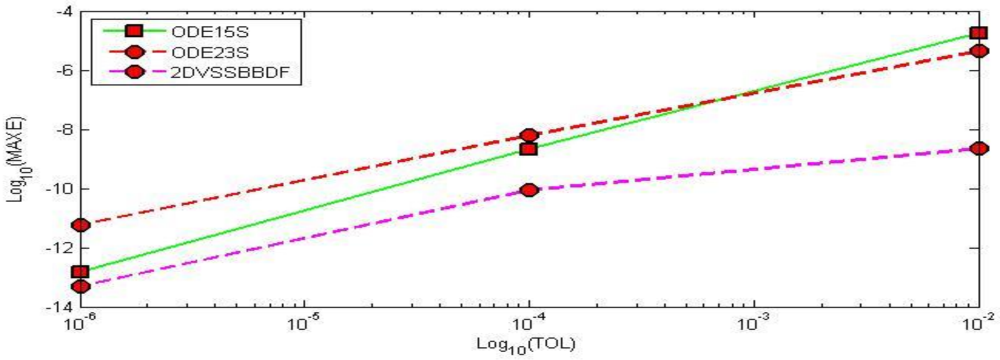

Figure 4: Accuracy Graph for Problem 1

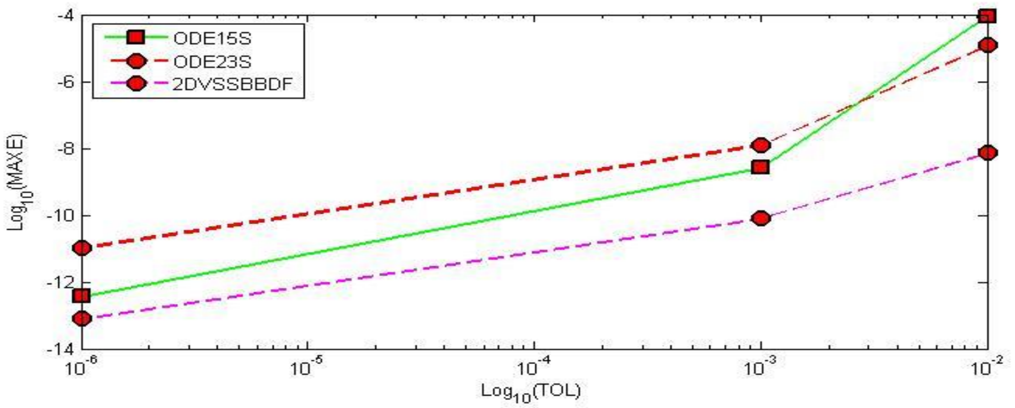

Figure 5: Accuracy Graph for Problem 2

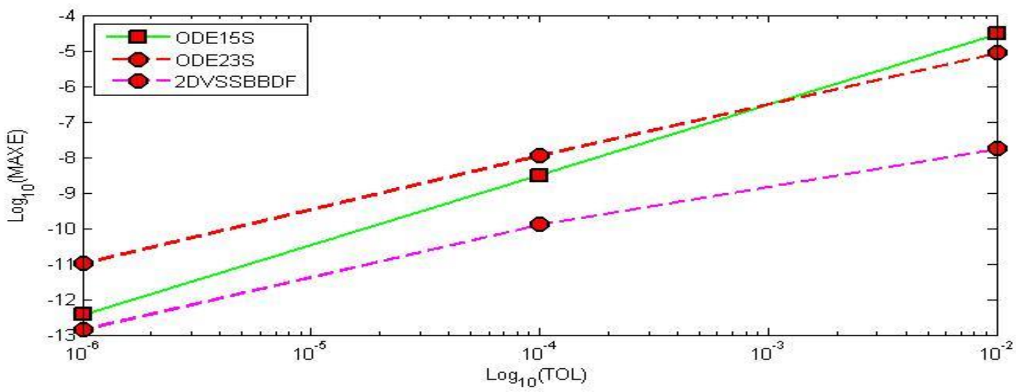

Figure 6: Accuracy Graph for Problem 3

RESULTS AND DISCUSSION

The table presents a performance comparison between ODE15s, ODE23s, and the proposed 2DVSSBBDF method across three decreasing tolerance levels 10^{-2}, 10^{-4}, and 10^{-6}. The comparison is based on the number of time steps (TS), the maximum error (MAXE), and the CPU execution time.

i. Accuracy (MAXE): The 2DVSSBBDF method consistently achieves lower maximum errors across all tolerance levels, demonstrating superior accuracy. At TOL 10^{-2}, its error is two orders of magnitude smaller than both ODE15s and ODE23s. Even at 10^{-6}, it maintains the lowest error, indicating better precision for stiff problems.

ii. Efficiency (CPU Time): The proposed method also exhibits significantly faster execution times. At all tolerance levels, 2DVSSBBDF requires a fraction of the CPU time compared to the MATLAB solvers, particularly outperforming ODE23s, which is slower and less accurate.

iii. Time Steps (TS): Although 2DVSSBBDF may use more steps than ODE23s at some tolerances, the overall computational cost remains lower due to its low per-step cost and efficient adaptive step size control.

CONCLUSION

A new numerical method called a new 2-point diagonally implicit variable step size super class of block backward differentiation formula for solving first order stiff IVPs is developed. The method computes two solution values per step and it is zero stable for the three different values of r. For A-stability property, analysis showed that the method is A-stable when $r = 2$, and almost A- stable when $r = 1$ and $r = \frac{5}{8}$ and its of order 4. The method was tested by solving some set of stiff IVPs. Numerical comparison is also made between the new method and some existing methods where the new method outperformed the two ode MATLAB Solvers namely ode15s and ode23s both in terms of accuracy and computation time

REFERENCES

Alhassan B. and Musa H. (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.* **12** : 101-124.

Cash J.R. (1980). On the integration of stiff systems of ODEs using Extended Backward Differentiation Formula. *Numerical Mathematics* **34**. 235-246.

Dahlquist, G. (1978). On Accuracy and Unconditional Stability of Linear Multistep Methods for Second Order Differential Equations, *BIT Numerical of Mathematics*, **18:** 133–136.

Gear, C. W (1965). Hybrid Multistep Methods for Initial Value Problems in Ordinary differential equations

Lambert J.D.(1991).Numerical methods for ordinary differential systems.New York: John Wiley.

Musa H., Suleiman M.B., & Senu N. (2012). A-stable 2-points Block Extended Backward Differentiation Formulae for solving ODEs. AIP Conference Proceedings **1450**:254-258.

Musa H., Suleiman M.B. and Ismail F. (2015). An Implicit 2-point Block Extended Backward Differentiation Formulas for solving Stiff IVPs. *Malaysian Journal of Mathematical Sciences*. **9**: 33-51.

Musa H. and Bala N.(2019). 3-Point Diagonally Implicit Super Class of Block Backward Differentiation Formula for solving Stiff Initial Value Problems. *Dutse Journal of Pure and Applied Sciences*. (1b): 1-10.

Musa H. and Unwala M.A. (2019). Extended 3-point Super Class of Block Backward
Differentiation Formula for solving Stiff IVP's. *Abacus (Mathematics Science series)*. **44**(1): 584-59.

Majid Z. A. and Suleiman M.B. (2006). Performance of 4-point diagonally implicit block method for solving first order stiff ODEs. *Department of Mathematics UTM*. **222**: 137-146.

Musa H. and Yusuf H. (2021). Derivation of Diagonally Implicit Extended 3-point Super class of Block Backward Differentiation Formula For Solving Stiff Initial Value Problems. 39th Annual Conference of the Nigerian Mathematical Society (NMS) at Redeemer's University Ede, Osun State, Nigeria.

Suleiman M. B., Musa H., Ismail F. and Senu N. (2013). A new variable step size block backward differentiation formula for solving stiff IVPs. *International Journal of Computer Mathematics* **90** (11): 2391 – 2408.

Suleiman M.B., Musa H., Ismail F., Senu N. and Ibrahim Z.B. (2014). A new super class of block backward differentiation formulas for stiff ODEs, *Asian European journal of mathematics* **7**:2291-2361

Yatim S. A. M., Ibrahim Z. B., Othman K. I and Suleiman M.B, (2011). A Quantitative Comparison of Numerical Method for Solving Stiff Ordinary Differential Equations. Hindawi Publishing Corporation, Journal of Mathematical Problems in Engineering. Article ID 193691, dio:10.1155/2011/193691, 1-12.

Yusuf H., Musa H. and Alhassan B. (2024): A New Fixed Coefficient Diagonally Implicit Block Backward Differentiation Formula for Solving Stiff Initial Value Problems. *UMYU Scientifica,* **3**(1), 1-14. https://doi.org/10.56919/usci.2431.001