UNIVERSITI TEKNOLOGI MARA

5th I-CReST 2025

INTERNATIONAL CONFERENCE ON RESEARCH AND PRACTICES IN SCIENCE, TECHNOLOGY AND SOCIAL SCIENCES

SYNERGY FOR FUTURE SUSTAINABILITY:
BRIDGING SCIENCE, TECHNOLOGY AND HUMANITIES

Proceeding Book

UNIVERSITI TEKNOLOGI MARA
CAWANGAN SELANGOR, KAMPUS DENGKIL
MALAYSIA

13th SEPTEMBER 2025

ORGANISED BY: Centre of Foundation Studies

STRATEGIC PARTNER: UiTM Global

IN COLLABORATION WITH: UTM

I-CReST 2025:150-111 – Singly Diagonally Implicit Block BDF of Order 3 for Stiff Ordinary Differential Equations

Hazizah Mohd Ijam[1]*, Nur Syafiqah Musa[1], Nor Syafiqah Sunwandi[1], Saufianim Jana Aksah[2], Norshakila Abd Rasid[3]

[1]*Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450, Shah Alam, Selangor, Malaysia.*
[2]*Centre of Foundation Studies, Universiti Teknologi MARA, Cawangan Selangor, Kampus Dengkil, 43800 Dengkil, Selangor, Malaysia.*
[3]*Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Perak Branch, Tapah Campus, 35400 Tapah Road, Perak, Malaysia.*

*E-mail: hazizahijam@uitm.edu.my

ABSTRACT

This paper presents the development of a Singly Diagonally Implicit Block Backward Differentiation Formula (SDIBBDF) of order 3 for solving stiff Ordinary Differential Equations (ODEs). The general formula is derived using Taylor series expansion, yielding two approximate solutions that are computed simultaneously in a block. The proposed method is shown to be zero-stable, consistent, convergent and *A*-stable. Numerical results based on maximum error and execution time, indicate that the proposed method outperforms the existing methods in solving stiff ODEs.

Keywords: Block backward differentiation formula; singly diagonally implicit; stiff ODEs

1. INTRODUCTION

First-order ordinary differential equations (ODEs) involve derivatives with respect to a single independent variable and one or more dependent variables. They are fundamental in mathematics and the sciences, as they describe the relationship between a function and its derivative with respect to one variable. ODEs are widely used to model various physical processes, including those in mechanics, chemistry and electrical circuits. A first order ODE, for example, is typically expressed as

$$y'(x) = f(x, y), y(a) = y_0, x \in [a, b] \quad (1)$$

where y is the dependent variable, x is the independent variable and $f(x, y)$ is a given function.

Numerous numerical methods have been developed to solve ODEs, including both linear and nonlinear types, as well as stiff and non-stiff cases. Among these, stiff ODEs are particularly challenging because some components of the solution change rapidly over short intervals, while others change much more slowly. This contrast requires the use of specialized numerical techniques that are better suited to handle stiffness. In many real-world mathematical models, the associated initial value problems (IVPs) are often stiff, particularly in areas such as vibration analysis, chemical engineering and control theory. Consequently, researchers continue to explore innovative analytical and numerical methods to effectively solve stiff IVPs in these complex systems.

The backward differentiation formula (BDF) has long been a widely used method for solving stiff ODEs. Traditionally, most numerical methods generate only a single

approximation of y_{n+1} at each step x_{n+1}. However, a study by Ibrahim et al. [1] introduced the block backward differentiation formula (BBDF) as an alternative approach aimed at reducing the number of integration steps and computational time required by conventional solvers, while preserving accuracy and satisfying stability requirements. This method has attracted considerable interest within the research community, demonstrating improved efficiency and accuracy over non-block methods and existing solvers, see [2] – [11]. Various studies have since applied the BBDF approach to stiff problems, further confirming its potential for enhancing both computational performance and solution reliability.

Subsequent research has improved the efficiency and accuracy of the BBDF method for solving stiff problems by incorporating a diagonally implicit structure. This structure reduces the computational effort required for evaluating differentiation coefficients, resulting in lower cumulative errors compared to fully implicit methods, see [3], [5] – [7] and [9] – [11]. In diagonally implicit methods, the system matrix is structured as a lower triangular matrix with constant diagonal entries, which simplifies the computational process. Notably, this allows the Jacobian matrix to be evaluated only once per step, thereby significantly reducing the computational load typically associated with fully implicit methods. This development has led to the formulation of singly diagonally implicit methods, where the matrix maintains identical constants along the diagonal, as demonstrated by Aksah et al. [5] and Aksah et al. [6], further minimizing computational cost.

Building upon this foundation, the present paper introduces the ρ-Singly Diagonally Implicit Block Backward Differentiation Formula (ρ-SDIBBDF), a refined extension of the BBDF method designed to offer improved performance and competitiveness with existing approaches.

2. DERIVATION OF THE METHOD

This section presents a detailed derivation of the ρ-SDIBBDF method. The operational procedure of the method is illustrated in Fig. 1.

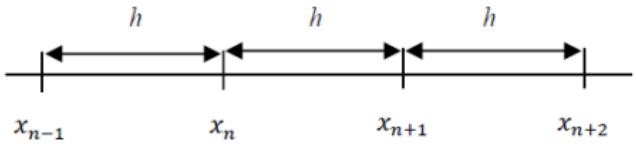

Fig. 1. 2-point block method of constant step size

The block method shown in Fig. 1 calculates approximate values for y_{n+1} and y_{n+2} simultaneously, utilizing a preceding block containing two previous points, x_{n-1} and x_n. In this paper, we developed the extended method in [7] to obtain the corrector formula of ρ-SDIBBDF method in the following form for a constant step size, h

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

where $\alpha_{k,k} = 1$, $\beta_{k-1,k} = \rho\beta_{k,k}$, $m = 3$ represents the order of the method and $k = 1, 2$ corresponds to the solution points y_{n+1} and y_{n+2}, respectively, with $\alpha_{ii} = \gamma$. The linear difference operator for the block multistep method, as shown in (2), is given by

$$\mathcal{L}_k[y(x_n);h] = \sum_{i=0}^{w} \left(\alpha_{k,i-2}\, y(x+(i-2)h)\right) - h\beta_{k,w}\,[y'(x+kh) - \rho y'(x+(k-1)h)] = C_q y^q + O(h^{p+1}). \tag{3}$$

where $w = k + m - 1$.

To obtain y_{n+1} and y_{n+2}, $k = i = 1{,}2$ are substituted accordingly in (2). Each point will be indicated by letting $\alpha_{k,k} = 1$ and considering $\alpha_{0,1} = 0$, $\alpha_{0,2} = 0$ for y_{n+1} and y_{n+2}, respectively. The Taylor's series expansion of (3) is solved concurrently yield the corresponding formula for ρ-SDIBBDF where $\rho = -0.75$ and takes the following form

$$\begin{aligned} y_{n+1} &= \frac{9}{25}hf_n + \frac{12}{25}hf_{n+1} + \frac{63}{50}y_n + \frac{1}{10}y_{n-2} - \frac{9}{25}y_{n-1}, \\ y_{n+2} &= \frac{9}{25}hf_{n+1} + \frac{12}{25}hf_{n+2} - \frac{9}{25}y_n + \frac{1}{10}y_{n-1} + \frac{63}{50}y_{n+1}. \end{aligned} \tag{4}$$

Equation (4) can be written in matrix form as

$$\begin{aligned} &\begin{bmatrix} 1 & 0 \\ -\frac{63}{50} & 1 \end{bmatrix} \begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix} \\ &\quad = \begin{bmatrix} 0 & \frac{1}{10} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_{n-3} \\ y_{n-2} \end{bmatrix} + \begin{bmatrix} -\frac{9}{25} & \frac{63}{50} \\ \frac{1}{10} & -\frac{9}{25} \end{bmatrix} \begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix} + h \begin{bmatrix} 0 & \frac{9}{25} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_n \end{bmatrix} \\ &\quad + h \begin{bmatrix} \frac{12}{25} & 0 \\ \frac{9}{25} & \frac{12}{25} \end{bmatrix} \begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix}. \end{aligned} \tag{5}$$

By taking $y' = \lambda y$ and insert $\lambda y = H$ into (5), then we have

$$\begin{bmatrix} 1 - \frac{12}{25}H & 0 \\ -\frac{63}{50} - \frac{9}{25}H & 1 - \frac{12}{25}H \end{bmatrix} \begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix} = \begin{bmatrix} -\frac{9}{25} & \frac{63}{50} + \frac{9}{25}H \\ \frac{1}{10} & -\frac{9}{25} \end{bmatrix} \begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix} + \begin{bmatrix} 0 & \frac{1}{10} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_{n-3} \\ y_{n-2} \end{bmatrix}, \tag{6}$$

which is equivalent to $PY_m = QY_{m-1} + RY_{m-2}$, where

$$P = \begin{bmatrix} 1 - \frac{12}{25}H & 0 \\ -\frac{63}{50} - \frac{9}{25}H & 1 - \frac{12}{25}H \end{bmatrix}, Q = \begin{bmatrix} -\frac{9}{25} & \frac{63}{50} + \frac{9}{25}H \\ \frac{1}{10} & -\frac{9}{25} \end{bmatrix}, R = \begin{bmatrix} 0 & \frac{1}{10} \\ 0 & 0 \end{bmatrix}$$

$$Y_m = \begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix}, Y_{m-1} = \begin{bmatrix} y_{n-1} \\ y_n \end{bmatrix}, Y_{m-2} = \begin{bmatrix} y_{n-3} \\ y_{n-2} \end{bmatrix}.$$

The constant C_q in (3) is given by

$$C_q = \sum_{j=0}^{k} \frac{j^q \alpha_j}{q!} - \frac{j^{(q-1)} \beta_j}{(q-1)!}, q = 0,1,2, \dots, p+1. \tag{7}$$

As stated in [11], the difference operator in (3) is deemed to be of order p if the coefficients $C_0 = C_1 = \dots = C_p = 0, C_{p+1} \neq 0$, where C_{p+1} is identified as the error constant. By plugging the obtained values for a_j and b_j into (3), the result $C_0 = C_1 = C_2 = C_3 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, C_4 = \begin{bmatrix} -\frac{9}{100} \\ -\frac{9}{100} \end{bmatrix}$ is obtained. The condition $C_4 \neq 0$ confirms that the ρ-SDIBBDF is of order 3 and will hereafter be referred to as ρ-SDIBBDF(3).

3. STABILITY PROPERTIES OF THE METHOD

This section outlines the essential criteria for ensuring the convergence of linear multistep methods (LMMs), as defined in Definitions 1, 2 and 3, emphasizing the requirements of consistency, zero stability and A-stability (see [12]).

Definition 1. The LMM is said to be consistent if it has order $p \geq 1$. The method is consistent if and only if the following conditions are satisfied.

$$\sum_{j=0}^{k} \alpha_j = 0,$$
$$\sum_{j=0}^{k} j\alpha_j = \sum_{j=0}^{k} \beta_j.$$

Definition 2. The method is said to be zero stable if there is no root of the first characteristic polynomial having modulus greater than one and if every root with modulus one is simple.

Definition 3. A numerical method is A – stable if its region of absolute stability covers the entire of the negative half-plane $Re(h\lambda) < 0$.

Upon substituting the coefficients of P, Q, and R from (6) into the determinant formula, denoted as $|Pr^2 - Qr - R| = 0$, we derive a stability polynomial with the following structure.

$$R = t^4 - \frac{24}{25}t^4H - \frac{2169}{2500}t^3 + \frac{144}{625}t^4H^2 - \frac{783}{625}t^3H - \frac{153}{1250}t^2 - \frac{81}{625}t^3H^2 - \frac{9}{125}t^2H - \frac{1}{100}t. \tag{8}$$

For zero stability, we set $H = 0$ in (8) to obtain the first characteristic polynomial as follows

$$R = t^4 - \frac{2169}{2500}t^3 - \frac{153}{1250}t^2 - \frac{1}{100}t. \tag{9}$$

By solving (9) for t, gives $t = 0, -0.06620 - 0.07496I, -0.06620 + 0.07496I$ and $t = 1$. Since all $|t| \leq 1$, therefore by Definition 2, the method is zero stable.

The boundaries of the stability regions for $\rho = -0.75$ is determined by setting $r = e^{i\theta}$ in (8) and solving R for t. The plots of the complex H-plane for a range of $\theta \in [0, 2\pi]$ where $|t| < 1$ are shown in Fig. 2.

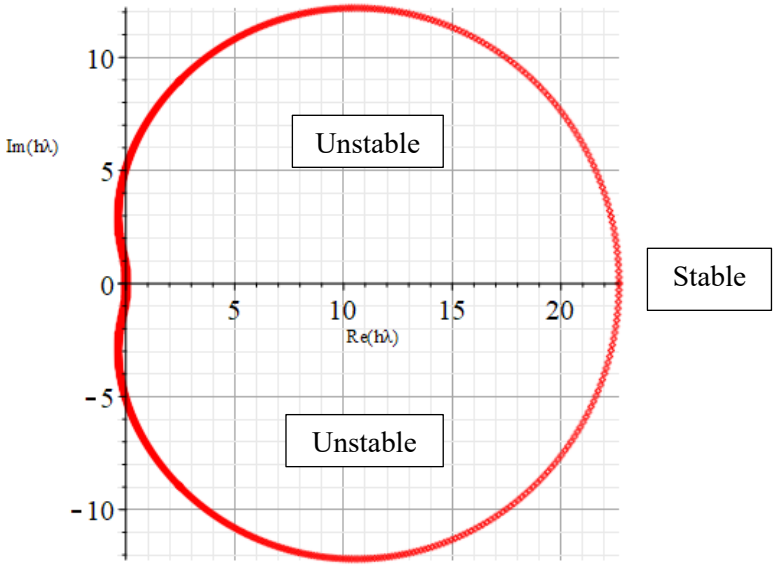

Fig. 2. Stability region for ρ-SDIBBDF(3) for $\rho = -0.75$

By observing the stability region in Fig. 2 and thus by Definition 3, ρ-SDIBBDF(3) is said to be A-stable.

According to Definition 1, the consistency of ρ-SDIBBDF(3) is assured since these methods have an order $p \geq 1$. Since ρ-SDIBBDF(3) demonstrate both consistency and zero stability, it can be concluded that the derived method is converged [13].

4. NUMERICAL RESULTS

Four test problems will be solved numerically using ρ-SDIBBDF(3). Each problem will be addressed using different step sizes, $h = 10^{-2}, 10^{-4}, 10^{-6}$.

Test Problem 1: Linear problem in [14]

$y' = 100(\sin(x) - y)$ $\quad y(0) = 0 \quad x \in [0, 3]$

Exact solution:

$$y(x) = \frac{1}{1.0001}(\sin x - 0.01\cos x + 0.01e^{-100x})$$

Eigenvalue: $\lambda = -100$

Test Problem 2: Nonlinear problem in [15]

$y'_1 = -(\varepsilon^{-1} + 2)y_1 + \varepsilon^{-1}{y_2}^2$ $\quad y_1(0) = 1 \quad x \in [0,20]$

$y'_2 = y_1 - y_2(1 + y_2)$ $\quad y_2(0) = 1$

Exact solutions:

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

Eigenvalues: $\lambda = -1, -100002$

Test Problem 3: Linear problem in [16]

$$y'_1 = -0.1y_1 \qquad y_1(0) = 1 \qquad x \in [0, 10]$$
$$y'_2 = -10y_2 \qquad y_2(0) = 1$$
$$y'_3 = -100y_3 \qquad y_3(0) = 1$$
$$y'_4 = -1000y_4 \qquad y_4(0) = 1$$

Exact solutions:

$$y_1(x) = e^{-0.1x}, y_2(x) = e^{-10x}, y_3(x) = e^{-100x}, y_4(x) = e^{-1000x}$$

Eigenvalues: $\lambda = -0.1, -10, -100, -1000$

Test Problem 4: Linear Problem in [17]

$$y'_1 = -21y_1 + 19y_2 - 20y_3 \qquad y_1(0) = 1 \qquad x \in [0, 10]$$
$$y'_2 = 19y_1 - 21y_2 + 20y_3 \qquad y_2(0) = 0$$
$$y'_3 = 40y_1 - 40y_2 - 40y_3 \qquad y_3(0) = -1$$

Exact solutions:

$$y_1(x) = \frac{1}{2}[e^{-2x} + e^{-40x}(\cos 40x + \sin 40x)]$$
$$y_2(x) = \frac{1}{2}[e^{-2x} - e^{-40x}(\cos 40x + \sin 40x)]$$
$$y_3(x) = 2e^{-40x}\left[-\frac{1}{2}\cos 40x + \frac{1}{2}\sin 40x\right]$$

Eigenvalues: $\lambda = -2, -40$

The numerical results for Test Problems 1–4 are compiled in Tables 1–4, accordingly. All the tested problems have been solved numerically by generating two approximations in a single block by using C++ Software. ρ-SDIBBDF(3) being compared with some existing methods of the same order in the literature. The numerical results of the maximum error and execution time are given in Tables 1–4, with

$$MAXE = \underbrace{max}_{1 \le i \le TS} \left(\underbrace{max}_{1 \le i \le N} |(y_i)_t - \big(y(x_i)\big)_t| \right)$$

where TS is the total number of steps, N is the number of equations, y_i and $y(x_i)$ are the approximated and exact solutions, respectively.

The following notations are used in the tables and figures below.

h	Step Size
TS	Total Steps
MAXE	Maximum Error
TIME	Execution Times in Microsecond
ρ-SDIBBDF(3)	The proposed method
ρ-DIBBDF(3)	ρ-Diagonally Implicit Block Backward Differentiation Formula of Order 3 in [7]
BBDF(3)	Block Backward Differentiation Formula of Order 3 in [1]
DIBBDF(3)	Diagonally Implicit Block Backward Differentiation Formula of Order 3 in [3]

Table 1. Numerical results for Test Problem 1

h	METHOD	ρ	TS	MAXE	TIME
10^{-2}	ρ-SDIBBDF(3)	-0.75	150	1.82796e-04	1.02328e-05
	ρ–DIBBDF(3)	-0.75	150	1.82796e-04	1.44938e-05
	BBDF(3)		150	7.32490e-04	3.07425e-04
	DIBBDF(3)		150	6.28527e-03	5.91831e-04
10^{-4}	ρ-SDIBBDF(3)	-0.75	15000	1.52831e-06	2.27680e-04
	ρ–DIBBDF(3)	-0.75	15000	1.52955e-06	2.46533e-04
	BBDF(3)		15000	7.18301e-05	1.09631e-03
	DIBBDF(3)		15000	1.72331e-06	1.83674e-03
10^{-6}	ρ-SDIBBDF(3)	-0.75	1500000	1.57948e-10	2.17504e-02
	ρ–DIBBDF(3)	-0.75	1500000	1.59675e-10	4.21601e-02
	BBDF(3)		1500000	7.35563e-07	8.01481e-02
	DIBBDF(3)		1500000	1.73616e-10	6.05115e-02

Table 2. Numerical results for Test Problem 2

h	METHOD	ρ	TS	MAXE	TIME
10^{-2}	ρ-SDIBBDF(3)	-0.75	1000	5.16894e-04	2.15233e-05
	ρ–DIBBDF(3)	-0.75	1000	5.16894e-04	4.16859e-05
	BBDF(3)		1000	8.30093e-03	4.85253e-04
	DIBBDF(3)		1000	7.46796e-04	7.40845e-04
10^{-4}	ρ-SDIBBDF(3)	-0.75	100000	6.30680e-08	1.58326e-04
	ρ–DIBBDF(3)	-0.75	100000	6.37046e-08	2.95505e-04
	BBDF(3)		100000	8.90434e-05	5.69954e-03
	DIBBDF(3)		100000	6.94730e-08	4.42888e-03
10^{-6}	ρ-SDIBBDF(3)	-0.75	10000000	1.10599e-11	1.73860e-02
	ρ–DIBBDF(3)	-0.75	10000000	2.39278e-11	2.32923e-02
	BBDF(3)		10000000	8.91027e-07	7.65485e-01
	DIBBDF(3)		10000000	2.30985e-11	1.73304e+00

Table 3. Numerical results for Test Problem 3

h	METHOD	ρ	TS	MAXE	TIME
10^{-2}	ρ-SDIBBDF(3)	-0.75	500	2.88931e+02	1.26317e-05
	ρ–DIBBDF(3)	-0.75	500	2.88931e+02	3.73198e-05
	BBDF(3)		500	3.34010e+03	3.24788e-04
	DIBBDF(3)		500	1.75959e+03	1.43185e-04
10^{-4}	ρ-SDIBBDF(3)	-0.75	50000	1.12590e-02	1.50521e-03
	ρ–DIBBDF(3)	-0.75	50000	1.12590e-02	2.63057e-03
	BBDF(3)		50000	5.67155e-02	1.44454e-02
	DIBBDF(3)		50000	1.66455e-02	1.68095e-02
10^{-6}	ρ-SDIBBDF(3)	-0.75	5000000	1.57476e-06	1.59499e-02
	ρ–DIBBDF(3)	-0.75	5000000	1.59054e-06	8.65868e-02
	BBDF(3)		5000000	7.34012e-04	2.19193e-01
	DIBBDF(3)		500000	1.73430e-06	7.22086e-01

Table 4. Numerical results for Test Problem 4

h	METHOD	ρ	TS	MAXE	TIME
10^{-2}	ρ-SDIBBDF(3)	-0.75	500	1.45990e-01	1.37232e-05
	ρ–DIBBDF(3)	-0.75		1.45990e-01	3.69631e-05
	BBDF(3)			1.14580e+25	2.03291e-04
	DIBBDF(3)			6.08664e-01	1.34293e-04
10^{-4}	ρ-SDIBBDF(3)	-0.75	50000	5.05522e-05	2.73665e-04
	ρ–DIBBDF(3)	-0.75		5.11045 e-05	4.29651e-04
	BBDF(3)			8.16801e-03	2.37062e-03
	DIBBDF(3)			5.55654e-05	1.06007e-03
10^{-6}	ρ-SDIBBDF(3)	-0.75	5000000	5.05600e-09	1.20465e-02
	ρ–DIBBDF(3)	-0.75		5.11183e-09	2.61469e-02
	BBDF(3)			8.22481e-05	6.33621e-01
	DIBBDF(3)			5.55636e-09	5.27749e-01

Fig. 3–6 show performance curves of ρ-SDIBBDF(3), ρ–DIBBDF(3), BBDF(3) and DIBBDF(3), by plotting their log_{10} MAXE against log_{10} TIME.

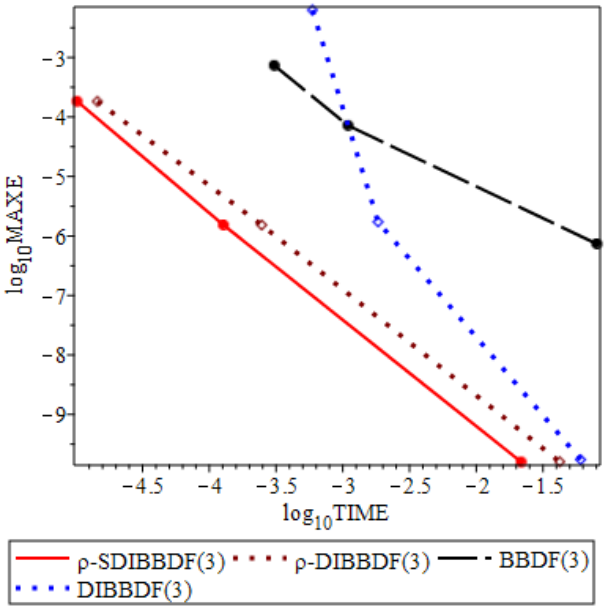

Fig. 3. Graph of log_{10} MAXE against log_{10} TIME for Test Problem 1

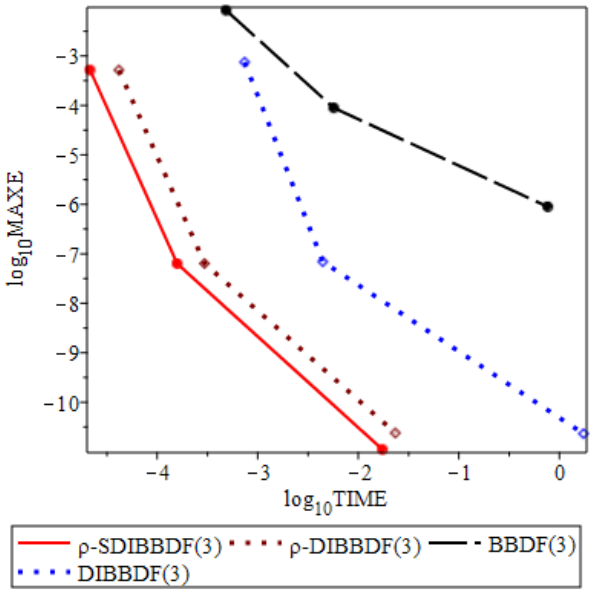

Fig. 4. Graph of log_{10} MAXE against log_{10} TIME for Test Problem 2

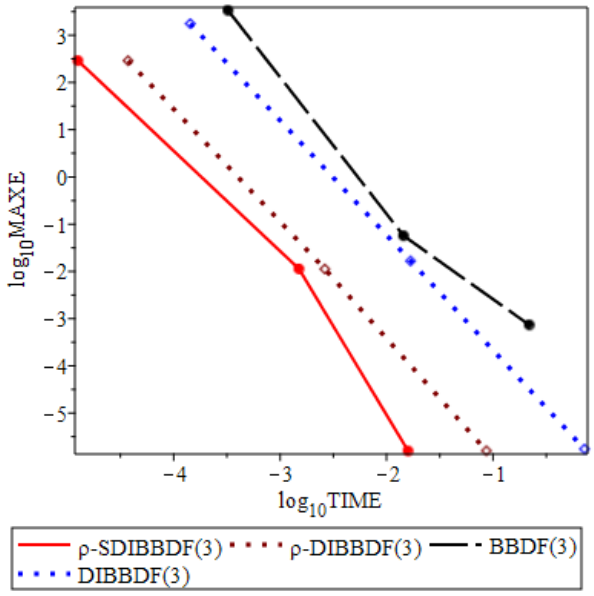

Fig. 5. Graph of log_{10} MAXE against log_{10} TIME for Test Problem 3

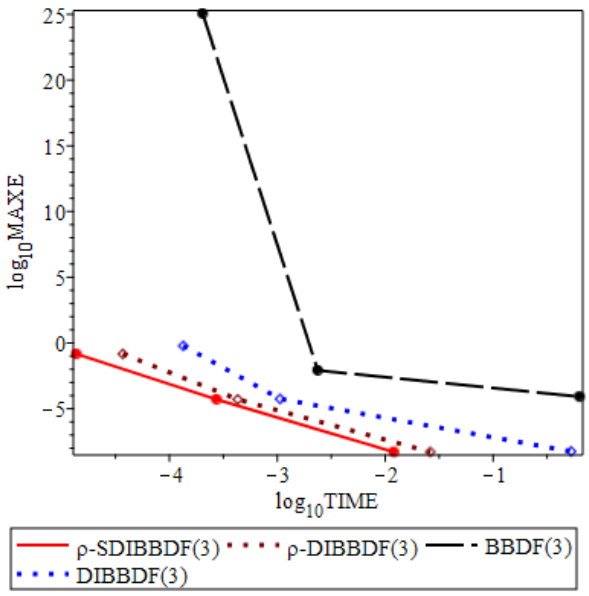

Fig. 6. Graph of log_{10} MAXE against log_{10} TIME for Test Problem 4

The performance curves of the methods ρ-SDIBBDF(3), ρ-DIBBDF(3), BBDF(3) and DIBBDF(3) reveal interesting insights into their efficiency. By plotting $\log_{10}$ MAXE and against $\log_{10}$ TIME for Test Problems 1 through 4, these methods are compared in terms of accuracy and computational time. The ρ-SDIBBDF(3) method consistently shows the best performance by achieving the lowest maximum error for a given computational time across different step sizes. The ρ-DIBBDF(3) method also performs well but it is slightly less accurate than ρ-SDIBBDF(3). On the other hand, the BBDF(3) method has the highest MAXE values, indicating lower accuracy despite using comparable or even more computational time. The DIBBDF(3) method strikes a balance between time and efficiency which performs moderately well.

Overall, ρ-SDIBBDF(3) consistently delivers the best results across all problems with ρ-DIBBDF(3) following closely. The BBDF(3) method shows variable performance that occasionally matching with ρ-DIBBDF(3), while DIBBDF(3) is generally the least efficient. This analysis underscores the robust performance of the ρ-based methods, especially ρ-SDIBBDF(3), across a range of numerical challenges.

5. CONCLUSION

In conclusion, the proposed ρ-SDIBBDF(3) method demonstrates superior accuracy and efficiency in solving stiff ODEs compared to existing methods such as ρ-DIBBDF(3), BBDF(3) and DIBBDF(3). It is proven to be zero-stable, consistent, convergent and A-stable. The inclusion of the free parameter ρ = -0.75 significantly enhances its performance, consistently yielding the lowest errors across test problems. Numerical results show that as the step size decreases, the method achieves lower maximum errors, confirming its precision.

To further improve the method, future work may focus on optimizing computational efficiency, incorporating adaptive step size control, applying it to a wider range of real-world stiff problems, and conducting detailed stability and error analyses. These efforts will further enhance the power and versatility of the ρ-SDIBBDF(3) method for solving stiff ODEs.

ACKNOWLEDGEMENT

This research was funded by MyRA Research Grant (MYRA-LPHD 2023) 600-RMC/GPM LPHD 5/3 (055/2023) from Universiti Teknologi MARA (UiTM). The authors also gratefully acknowledge to School of Mathematical Sciences, Faculty of Computer and Mathematical Sciences, UiTM for supporting this research.

REFERENCES

[1] Z. B. Ibrahim, K. I. Othman and M. B. Suleiman, "Implicit *r*-point block backward differentiation formula for solving first-order stiff ODEs," *Applied Mathematics and Computation*, vol. 186, no. 1, pp. 558–565, 2007.

[2] M. B. Suleiman, H. Musa, F. Ismail and N. Senu, "A new variable step size block backward differentiation formula for solving stiff initial value problems," *International Journal of Computer Mathematics*, vol. 90, no. 11, pp. 2391– 2408, 2013.

[3] I. S. M. Zawawi, Z. B. Ibrahim and K. I. Othman, "Derivation of diagonally implicit block backward differentiation formulas for solving stiff initial value problems," *Mathematical Problems in Engineering*, vol. 2015, no. 1, pp. 1–13, 2015.

[4] A. F. N. Rasedee, H. M. Ijam, M. H. A. Sathar, N. Ishak, S. R. Hamzah, M. Sahrim and I. Ismail, "Solution for nonlinear Riccati equation by block method," *AIP Conference Proceedings*, vol. 1974, no. 1, 2018.

[5] S. J. Aksah, Z. B. Ibrahim and I. S. M. Zawawi, "Stability analysis of singly diagonally implicit block backward differentiation formulas for stiff ordinary differential equations," *Mathematics,* vol. 7, no. 2, pp. 211, 2019a.

[6] S. J. Aksah and Z. B. Ibrahim, "Singly diagonally implicit block backward differentiation formulas for HIV infection of CD4+T cells," *Symmetry*, vol. 11, no. 5, pp. 625, 2019b.

[7] H. M. Ijam and Z. B. Ibrahim, "Diagonally implicit block backward differentiation formula with optimal stability properties for stiff ordinary differential equations," *Symmetry*, vol. 11, no. 11, pp. 1342, 2019.

[8] I. S. M. Zawawi and Z. B. Ibrahim, "BBDF-α for solving stiff ordinary differential equations with oscillating solutions," *Tamkang Journal of Mathematics*, vol. 51, no. 2, pp. 123–136, 2020.

[9] H. M. Ijam, Z. B. Ibrahim, Z. A. Majid and N. Senu, "Stability analysis of a diagonally implicit scheme of block backward differentiation formula for stiff pharmacokinetics models," *Advances in Difference Equations*, vol. 2020: 400, 2020.

[10] N. Zainuddin, Z. B. Ibrahim and I. S. M Zawawi, "Diagonal block method for stiff Van der Pol equation," *IAENG International Journal of Applied Mathematics*, vol. 53, no. 1, pp. 1–8, 2023.

[11] Z. B. İbrahim, H. M. Ijam, S. J. Aksah and N. A. Rasid, "Enhancing accuracy and efficiency in stiff ODE integration using variable step diagonal BBDF approaches," *Malaysian Journal of Fundamental and Applied Sciences*, vol. 20, no 5, pp. 1083–1100, 2024.

[12] J. D. Lambert, Computational Methods in Ordinary Differential Equations. New York: John Wiley & Sons, 1973.

[13] G. Hall and J. M. Watt. Modern Numerical Methods for Ordinary Differential Equations. Clarendon Press, 1976.

[14] G. G. Dahlquist, "A special stability problem for linear multistep methods," *BIT Numerical Mathematics*, vol. 3, no. 1, pp. 27–43, 1963.

[15] H. Aminikhah and M. Hemmatnezhad, "An effective modification of the homotopy perturbation method for stiff systems of ordinary differential equations," *Applied Mathematics Letters*, vol. 24, no. 9, pp. 1502–1508, 2011.
[16] W. H. Enright, "Second derivative multistep methods for stiff ordinary differential equations," *SIAM Journal on Numerical Analysis*, vol. 11, no. 2, pp. 321–331, 1974.
[17] R. K. Sahi, S. N. Jator and N. A. Khan, "A Simpson's-type second derivative method for stiff systems," *International Journal of Pure and Applied Mathematics*, vol. 81, no. 4, pp. 619–633, 2012.