ASM Sc. J., 12, Special Issue 1, 2019 for IQRAC2018, 33-40

A Numerical Solver for Second Order Stiff Ordinary Differential Equations

Asma Izzati Asnor*[1], Siti Ainor Mohd Yatim[1], and Zarina Bibi Ibrahim[2]

[1] *School of Distance Education, Universiti Sains Malaysia, 11800 USM, Pulau Pinang, Malaysia*

[2] *Faculty of Science, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*

* *Corresponding author:asma_asnor@yahoo.com*

This paper emphasizes on a new direct numerical solver of second order ordinary differential equations (ODEs). We developed this solver using block backward differentiation formula (BBDF) method with strategy on step size selection. The solver is known as direct variable step block backward differentiation formula (DVS-BBDF2). The numerical experiments have been conducted on some problems to validate the efficiency of the proposed method in terms of accuracy, number of total steps and computational time.

Keywords: backward differentiation formulae, block backward differentiation formulae, ordinary differential equations, second order ordinary differential equations, stiff ordinary differential equations .

I. Introduction

Basically, ODE is classified into initial value problem (IVP) and boundary value problem (BVP). IVP is an ODE together with its initial conditions whereas BVP is an ODE with its boundary conditions. Broad study of the ODEs has been widely applied for mathematical modeling in biology, physics, engineering, and chemistry. This mathematical modeling transforms the problems from an application area into tractable mathematical formulations whose theoretical and numerical analysis provide insight, answers, and guidance useful for the originating application (**?**).

In this paper, we will be focusing on the second order stiff initial value problems (IVPs) where the general form of the second order ODE is written as

$$y'' = f(x, y, y'), \;\; y(a) = y_0, \;\; y'(a) = y'_0 \qquad (1)$$

where $x \in [a, z]$, a is an initial point and z is the end point.

Definition 1. *(Lambert, 1973)*
The problem in (1) is stiff if

(i) $Re(\lambda_t) < 0$, *t=1,2,...,m and*
(ii) $max_t \mid Re(\lambda_t) \mid >> min_t \mid Re(\lambda_t) \mid$ *where* λ_t *are the eigenvalues of the Jacobian matrix,* $\frac{df}{dy}$

Although there are many numerical methods that exist in the literature, most of them are not able to solve the second order problem directly without reducing to lower order equation. A numerical solution of the second order problem in (1) is obtained using the numerical methods that are designed for first order problem after it is converted into first order problem. However, these methods have been found to have some drawbacks such as computational burden, wastage of computer time and a lot of human effort (Abdelrahim and Omar, 2016). Furthermore, all of these methods also need a lot of computational effort when computing differentiation coefficients at every step.

To overcome the drawbacks, a direct approach is more preferable where the second order ODE can be solved directly and efficiently. By overcoming these, the computational cost and computational effort can be reduced. Currently, (Ibrahim et al., 2008) proposed an efficient direct block method based on backward differentiation formula for solving stiff ODEs which possess some advantages such as providing two solutions at each step, no repetitive calculation on differentiation coefficients, producing quicker execution time, and reducing the total number of steps. Block backward differentiation formula (BBDF) is proven to be efficient for solving first order problem (Asnor et al., 2018) and also second order problem (Ibrahim et al., 2008, Zainuddin et al., 2016) than reduction methods.

In this paper, we aim to propose an alternative solver of order three using variable step size approach for solving the second order stiff ODEs directly. The zero stability of this method is ensured and discussed in (Yatim, 2013). The development of the method is briefly explained in Section II.. Then, Section III. shows the implementation of the proposed method in Newton's iteration form. The numerical experiments are conducted in Section IV. to validate the performance of the proposed method by comparing the performance with two ODE's solver in MATLAB. Therefore, the results are discussed in the next section and we conclude the whole study in the last section.

II. Development of DVS-BBDF2 Method

The suggested method will be developed using three back values in the preceding block and by varying the step size (constant step size, half of the step size and increase the step size to a factor of 1.8). Besides, this method also has special ability by which it will produce two approximation solutions simultaneously at points y_{n+1} and y_{n+2} in the current block.

The whole derivation is done with the help from Maple software. The derivation begins by replacing the *f(x,y)* in the equation (1) with Lagrange polynomial, *P(x)* as follows

$$y''(x) = P(x) \tag{2}$$

The polynomial passes through the points, $((x_{n-2}, y_{n-2}), (x_{n-1}, y_{n-1}), (x_n, y_n), (x_{n+1}, y_{n+1}), (x_{n+2}, y_{n+2}))$ and gives

$$P(x) = \sum_{j=0}^{4} P_4(x) = \sum_{j=0}^{4} y(x_{n+2-j}) \cdot L_{4,j}(x) \tag{3}$$

Let $x_{n+1} = x - s \cdot h$ and substitute into (3) yields

$$\begin{aligned} P(s \cdot h + x_{n+1}) &= \sum_{j=0}^{4} P_4(s \cdot h + x_{n+1}) \\ &= \sum_{j=0}^{4} y(x_{n+2-j}) \cdot L_{4,j}(s \cdot h + x_{n+1}) \end{aligned} \tag{4}$$

Then, differentiating the polynomial (4) with respect to s twice will give the following equations.

First differentiation:

$$\begin{aligned} h \cdot P'(s \cdot h + x_{n+1}) &= h \cdot \sum_{j=0}^{4} P_4'(s \cdot h + x_{n+1}) \\ &= h \cdot \sum_{j=0}^{4} y(x_{n+2-j}) \cdot L_{4,j}'(s \cdot h + x_{n+1}) \end{aligned} \tag{5}$$

Second differentiation:

$$\begin{aligned} h^2 \cdot P''(s \cdot h + x_{n+1}) &= h^2 \cdot \sum_{j=0}^{4} P_4''(s \cdot h + x_{n+1}) \\ &= h^2 \cdot \sum_{j=0}^{4} y(x_{n+2-j}) \cdot L_{4,j}''(s \cdot h + x_{n+1}) \end{aligned} \tag{6}$$

To obtain the first point, substitute s with 0 into equations (5) and (6) yield the updated polynomials such that

$$hy'_{n+1} = h \cdot \sum_{j=0}^{4} P'_4(x_{n+1})$$

$$= h \cdot \sum_{j=0}^{4} y(x_{n+2-j}) \cdot L'_{4,j}(x_{n+1}) \qquad (7)$$

$$h^2 f_{n+1} = h^2 \cdot \sum_{j=0}^{4} P''_4(x_{n+1})$$

$$= h^2 \cdot \sum_{j=0}^{4} y(x_{n+2-j}) \cdot L''_{4,j}(x_{n+1}) \qquad (8)$$

Therefore, the coefficients for hy'_{n+1} and $h^2 f_{n+1}$ are written in the Table 1 after replacing the value of r with 1, 2 and 5/9 where r is the step size ratio.

For second point, the coefficients of hy'_{n+2} and $h^2 f_{n+2}$ are obtained by substituting s with 1 into (5) and (6). The polynomials then become

$$hy'_{n+2} = h \cdot \sum_{j=0}^{4} P'_4(x_{n+1})$$

$$= h \cdot \sum_{j=0}^{4} y(x_{n+2-j}) \cdot L'_{4,j}(x_{n+1}) \qquad (9)$$

$$h^2 f_{n+2} = h^2 \cdot \sum_{j=0}^{4} P''_4(h + x_{n+1})$$

$$= h^2 \cdot \sum_{j=0}^{4} y(x_{n+2-j}) \cdot L''_{4,j}(h + x_{n+1}) \qquad (10)$$

Therefore, the coefficients for hy'_{n+2} and $h^2 f_{n+2}$ are written in the Table 2 after replacing the value of r with 1, 2 and 5/9.

III. Implementation of DVS-BBDF2 Method

The formulae of DVS-BBDF2 method derived in the previous section written in matrix form has been implemented using Newton's iteration. Therefore, the systems of linear equations will be solved in this section.

It is noted that $(i+1)^{th}$ is the iterative values of y_{n+1} and y_{n+2} .The following equation is defined as follows:

$$\begin{bmatrix} e_{n+1} \\ e_{n+2} \end{bmatrix}^{i+1} = \begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix}^{i+1} - \begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix}^{i} \qquad (11)$$

The linear systems, $\hat{E} = \hat{M}^{-1}\hat{N}$ at three values of step size ratio are

$$\hat{E} = \begin{bmatrix} e_{n+1} \\ e_{n+2} \end{bmatrix}^{i+1},$$

$$\hat{M} = \begin{bmatrix} t & w \\ x & y \end{bmatrix},$$

$$t = 1 - \alpha_1 h^2 \left(\frac{\partial f_{n+1}}{\partial y_{n+1}}\right) - \alpha_1 \beta_1 h \left(\frac{\partial f_{n+1}}{\partial y'_{n+1}}\right),$$
$$w = -\theta_1 \left(\frac{\partial f_{n+1}}{\partial y_{n+1}}\right) - \alpha_1 \beta_2 h \left(\frac{\partial f_{n+1}}{\partial y'_{n+1}}\right),$$
$$x = -\theta_2 \left(\frac{\partial f_{n+1}}{\partial y_{n+2}}\right) - \alpha_2 \hat{\beta}_1 h \left(\frac{\partial f_{n+1}}{\partial y'_{n+2}}\right),$$
$$y = 1 - \alpha_2 h^2 \left(\frac{\partial f_{n+2}}{\partial y_{n+2}}\right) - \alpha_2 \hat{\beta}_2 h \left(\frac{\partial f_{n+2}}{\partial y'_{n+2}}\right),$$

$$\hat{N} = \begin{bmatrix} v + B_1 \\ q + B_2 \end{bmatrix}^{i},$$

$$v = -y_{n+1} + \theta_1 y_{n+2} + \alpha_1 h^2 f(y_{n+1}, y'_{n+1})$$
$$q = -y_{n+2} + \theta_2 y_{n+2} + \alpha_2 h^2 f(y_{n+2}, y'_{n+2})$$

B_1 =Backvalues for first point
B_2 =Backvalues for second point

$r = 1$:

Table 1: The coefficients for the first point

Step size ratio		y_{n-2}	y_{n-1}	y_n	y_{n+1}	y_{n+2}
r=1	hy'_{n+1}	$-\frac{1}{12}$	$\frac{1}{2}$	$-\frac{3}{2}$	$\frac{5}{6}$	$\frac{1}{4}$
	$h^2 f_{n+1}$	$-\frac{1}{12}$	$\frac{1}{3}$	$\frac{1}{2}$	$-\frac{5}{3}$	$\frac{11}{12}$
r=2	hy'_{n+1}	$-\frac{1}{80}$	$\frac{5}{48}$	$-\frac{15}{16}$	$\frac{8}{15}$	$\frac{5}{16}$
	$h^2 f_{n+1}$	$-\frac{1}{120}$	$\frac{1}{24}$	$\frac{7}{8}$	$-\frac{28}{15}$	$\frac{23}{24}$
$r=\frac{5}{9}$	hy'_{n+1}	$-\frac{729}{1900}$	$\frac{13851}{8050}$	$-\frac{133}{50}$	$\frac{297}{266}$	$\frac{19}{62}$
	$h^2 f_{n+1}$	$-\frac{6561}{13300}$	$\frac{6561}{4025}$	$-\frac{31}{50}$	$-\frac{185}{133}$	$\frac{563}{644}$

Table 2: The coefficients for the second point

Step size ratio		y_{n-2}	y_{n-1}	y_n	y_{n+1}	y_{n+2}
r=1	hy'_{n+2}	$\frac{1}{4}$	$-\frac{4}{3}$	3	-4	$\frac{25}{12}$
	$h^2 f_{n+2}$	$\frac{11}{12}$	$-\frac{14}{3}$	$\frac{19}{2}$	$-\frac{26}{3}$	$\frac{35}{12}$
r=2	hy'_{n+2}	$\frac{1}{30}$	$-\frac{1}{4}$	$\frac{3}{2}$	$-\frac{16}{5}$	$\frac{23}{12}$
	$h^2 f_{n+2}$	$\frac{7}{60}$	$-\frac{5}{6}$	$\frac{17}{4}$	$-\frac{88}{15}$	$\frac{7}{3}$
$r=\frac{5}{9}$	hy'_{n+2}	$\frac{16767}{13300}$	$-\frac{2916}{575}$	$\frac{161}{25}$	$-\frac{92}{19}$	$\frac{1425}{644}$
	$h^2 f_{n+2}$	$\frac{63423}{13300}$	$-\frac{74358}{4025}$	$\frac{1103}{50}$	$-\frac{1562}{133}$	$\frac{2183}{644}$

$\alpha_1 = -\frac{3}{5}, \alpha_2 = \frac{12}{35}, \beta_1 = \frac{5}{6}, \beta_2 = \frac{1}{4},$

$\hat{\beta}_1 = -4, \hat{\beta}_2 = \frac{25}{12}, \theta_1 = \frac{11}{20}, \theta_2 = \frac{104}{35},$

$B_1 = -\frac{1}{20} y_{n-2} + \frac{1}{5} y_{n-1} + \frac{3}{10} y_n,$

$B_2 = -\frac{11}{35} y_{n-2} + \frac{8}{5} y_{n-1} - \frac{114}{35} y_n$

$\boldsymbol{r = 2}$**:**

$\alpha_1 = -\frac{15}{28}, \alpha_2 = \frac{3}{7}, \beta_1 = \frac{8}{15}, \beta_2 = \frac{5}{16},$
$\hat{\beta}_1 = -\frac{16}{5}, \hat{\beta}_2 = \frac{23}{12}, \theta_1 = \frac{115}{224}, \theta_2 = \frac{88}{35},$

$B_1 = -\frac{1}{224}y_{n-2} + \frac{5}{225}y_{n-1} + \frac{15}{32}y_n,$

$B_2 = -\frac{1}{20}y_{n-2} + \frac{5}{14}y_{n-1} - \frac{51}{28}y_n$

$\boldsymbol{r = 5/9}$**:**

$\alpha_1 = -\frac{133}{185}, \alpha_2 = \frac{644}{2183}, \beta_1 = \frac{297}{266}, \beta_2 = \frac{1425}{644},$

$\hat{\beta}_1 = \frac{19}{92}, \hat{\beta}_2 = -\frac{92}{19}, \theta_1 = \frac{10697}{17020}, \theta_2 = \frac{143704}{41477},$

$B_1 = -\frac{6561}{18500}y_{n-2} + \frac{124659}{106375}y_{n-1} - \frac{4123}{9250}y_n,$

$B_2 = -\frac{1458729}{1036925}y_{n-2} + \frac{297432}{54575}y_{n-1} - \frac{355166}{54575}y_n$

IV. Numerical Experiments

Two problems of second order stiff ODEs are taken from (Ibrahim et al., 2008) to test the performance of the proposed method.

Problem 1
$y'' = -16y - 8y',$
$y(0) = 1, \;\; y'(0) = -12,$
$x \in [0, 10],$
$y(x) = e^{-4x}(1 - 8x)$

$Eigenvalues: -4, -4$

Problem 2
$y'' = -1000y - 70y',$
$y(0) = 2, \;\; y'(0) = -70,$
$x \in [0, 2],$
$y(x) = e^{-50x} + e^{-20x}$

$Eigenvalues: -20, -50$

V. Results and Discussions

This section presents the performance results of DVS-BBDF2. We discuss the performance of our method with order three BBDF2 method which has lower step size increment value from the proposed method. The performance is determined at three different values of error tolerances limit, $10^{-2}, 10^{-4}$, and 10^{-6} to validate which method gives better performance in terms of maximum errors, number of total steps taken and computational time. The results are illustrated in the Tables 3-4 and the Figures 1-6.

Define the maximum error as

$$MAXE = \max_{1<i<TS}(ME)$$

where $G = y_i - y(x_i), K = A + B(y(x_i)),$

$$ME = \max_{1<i<TS}(|\frac{G}{K}|),$$

TS is the total steps, n is the number of equations, A and B are equal to 1.

As to be clear, the notations used in the tables and figures are listed as follows.

MR	Maximum error
AR	Average error
TOLV	Tolerance limit
TIME	Computational time
NTS	Number of total steps for computating the approximate solution
DVS-BBDF2	Direct variable step block backward differentiation formula
BBDF2	Order three block backward differentiation formula

The results in Tables 3-4 are illustrated in the following figures.

From the final outcomes that are listed in the Tables 3-4, DVS-BBDF2 method is producing the least maximum errors than BBDF2 for all

Table 3: Performance of the methods for Problem 1

Method		10^{-2}	10^{-4}	10^{-6}
DVS-BBDF2	MR	5.3071e-004	2.7672e-005	1.1104e-006
	NTS	35	71	174
	TIME	0.003616	0.026238	0.127348
BBDF2	MR	1.8840e-003	1.1381e-004	4.5819e-006
	NTS	27	52	120
	TIME	0.001235	0.001442	0.001999

Table 4: Performance of the methods for Problem 2

Method		10^{-2}	10^{-4}	10^{-6}
DVS-BBDF2	MR	6.6169e-004	3.2545e-005	1.3569e-006
	NTS	34	67	174
	TIME	0.002137	0.026763	0.122039
BBDF2	MR	1.9115e-003	1.1411e-004	4.6212e-006
	NTS	26	51	129
	TIME	0.001136	0.001310	0.001898

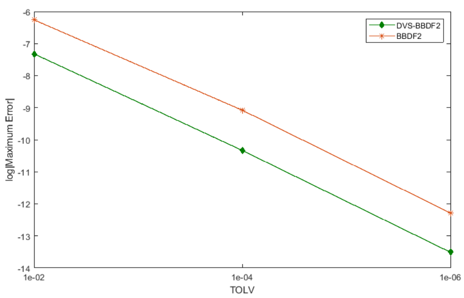

Figure 1: Performance in terms of maximum error for Problem 1

the problems considered in this research where it manages to reduce the error value. Besides, we can also see in the Figures 1 and 4 that all the maximum errors produced by DVS-BBDF2 method are within the tolerance value. For the outcome of total steps as illustrated in the Figures 2 and 5, DVS-BBDF2 method produceS more number of steps to compute the approximate solutions and Figures 3 and 6 show that DVS-BBDF2 need longer execution time than BBDF2 for all the problems.

VI. Conclusion

After the numerical experiments have gained the outcomes, we can conclude that DVS-BBDF2 manage to produce better accuracy than BBDF2 method. Increasing the value of step size will produce more accurate results but need more total steps and longer execution time. Besides, the proposed method is able to solve the second order ODEs directly by producing less maximum error which can produce more accurate results.

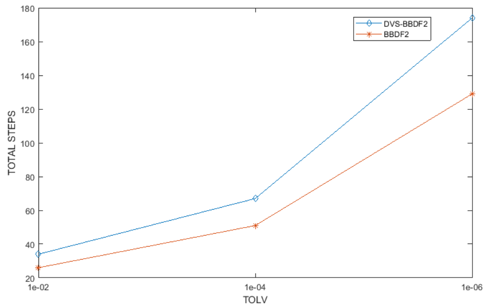

Figure 2: Performance in terms of number of total steps for Problem 1

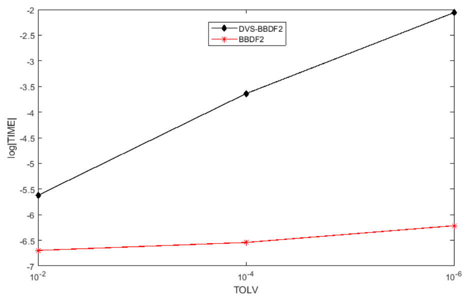

Figure 3: Performance in terms of computational time for Problem 1

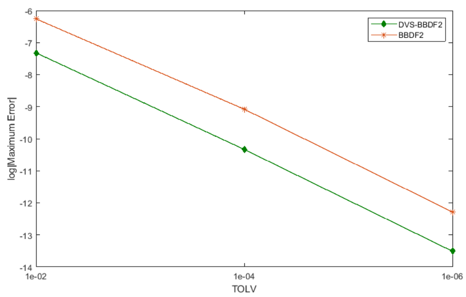

Figure 4: Performance in terms of maximum error for Problem 2

Acknowledgements

Thank you Universiti Sains Malaysia for supporting this project under USM short-term grant, 304/PJJAUH/6315189.

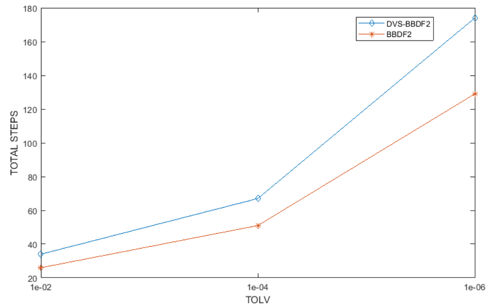

Figure 5: Performance in terms of number of total steps for Problem 2

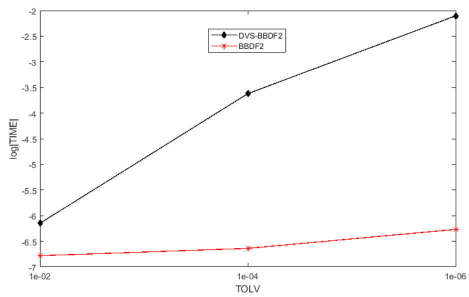

Figure 6: Performance in terms of computational time for Problem 2

References

[1] Ra'ft Abdelrahim and Zurni Omar. Derivation of a single-step hybrid block method with generalized two off-step points for solving second order ordinary differential equation directly. *International Journal of Mathematics and Computer in Simulation*, 10:171–179, 2016.

[2] Asma Izzati Asnor, Siti Ainor Mohd Yatim, and Zarina Bibi Ibrahim. Algorithm of Modified variable step block backward differentiation formulae for solving first order stiff odes. In *AIP Conference Proceedings*, volume 1974, pages 1–10. AIP Publishing, 2018.

[3] Zarina Bibi Ibrahim, Mohamed Suleiman, and Khairil Iskandar Othman. Direct block backward differentia-

tion formulas for solving second order ordinary differential equations. *International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering*, 2(4):260–262, 2008.

[4] Zarina Bibi Ibrahim, Mohamed Suleiman, and Khairil Iskandar Othman. Fixed coefficients block backward differentiation formulas for the numerical solution of stiff ordinary differential equations. *European Journal of Scientific Research*, 21(3):508–520, 2008.

[5] John David Lambert. *Computational Methods in Ordinary Differential Equations*. John Wiley & Sons, Inc, London, 1973.

[6] Siti Ainor Mohd Yatim. *Variable Step Variable Order Block Backward Differentiation Formulae for Solving Stiff Ordinary Differential Equations*. phdthesis, 2013.

[7] Nooraini Zainuddin, Zarina Bibi Ibrahim, Khairil Iskandar Othman, and Mohamed Suleiman. Direct fifth order block backward differentiation formulas for solving second order ordinary differential equations. *Chiang Mai J. Sci.*, 43(5):1171–1181, 2016.