Mathematics and Statistics 10(5): 942-955, 2022 http://www.hrpub.org
DOI: 10.13189/ms.2022.100506

Accuracy Improvement of Block Backward Differentiation Formulas for Solving Stiff Ordinary Differential Equations Using Modified Versions of Euler's Method

Nurfaezah Mohd Husin[1], Iskandar Shah Mohd Zawawi[1,*], Nooraini Zainuddin[2], Zarina Bibi Ibrahim[3]

[1]Faculty of Computer & Mathematical Sciences, Kompleks Al-Khawarizmi, Universiti Teknologi MARA, Selangor, Malaysia
[2]Mathematical and Statistical Sciences, Institute of Autonomous System, Department of Fundamental and Applied Sciences, Universiti Teknologi PETRONAS, Perak, Malaysia
[3]Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, Selangor, Malaysia

Received June 6, 2022; Revised August 16, 2022; Accepted August 30, 2022

Cite This Paper in the Following Citation Styles

(a): *[1] Nurfaezah Mohd Husin, Iskandar Shah Mohd Zawawi, Nooraini Zainuddin, Zarina Bibi Ibrahim , "Accuracy Improvement of Block Backward Differentiation Formulas for Solving Stiff Ordinary Differential Equations Using Modified Versions of Euler's Method," Mathematics and Statistics, Vol. 10, No. 5, pp. 942 - 955, 2022. DOI: 10.13189/ms.2022.100506.*

(b): *Nurfaezah Mohd Husin, Iskandar Shah Mohd Zawawi, Nooraini Zainuddin, Zarina Bibi Ibrahim (2022). Accuracy Improvement of Block Backward Differentiation Formulas for Solving Stiff Ordinary Differential Equations Using Modified Versions of Euler's Method. Mathematics and Statistics, 10(5), 942 - 955. DOI: 10.13189/ms.2022.100506.*

Abstract In this study, the fully implicit 2-point block backward differentiation formulas (BBDF) method has been successfully utilized for solving stiff ordinary differential equations (ODEs) by taking into account the uses of new starting methods namely, modified Euler's method (MEM), improved modified Euler's method (IMEM), and new Euler's method (NEM). The reason of proposing the BBDF is that the method has been proven useful for stiff ODEs due to its A-stable properties. Furthermore, the method is able to approximate the solutions at two points simultaneously at each step. The proposed method is also implemented through Newton's iteration procedure, which involves the calculation of the Jacobian matrix. Accuracy of the method is evaluated based on its performance in solving linear and non-linear initial value problems (IVPs) of first order stiff ODEs with transient and steady-state solutions. Some comparisons are made with the conventional BBDF approach for indicating the reliability of the proposed method. Numerical results indicate that not only classical Euler's method provides accurate solutions for BBDF, but also the numerous modified versions of Euler's methods improve the accuracy of BBDF, in terms of absolute error at certain step size and stage of iteration.

Keywords Block Method, Ordinary Differential Equations, Stiff, Euler's Method

1. Introduction

The ordinary differential equations (ODEs) involve ordinary derivatives of one or more dependent variables with respect to a single independent variable [1]. The initial value problems (IVPs) of first-order ODEs can be represented in the following form:

$$\frac{d\mathbf{y}}{dt} = y' = \mathbf{f}(\mathbf{y}, t), \quad y(0) = \mathbf{y}_0, \tag{1}$$

where the vector of variables, $\mathbf{y}(t) = (y_1(t), y_1(t), \ldots, y_n(t)) \in \mathbb{R}^n$, describes n properties of interest in system, evolving for times $t \geq 0$ starting from the initial value $y(0) = \mathbf{y}_0$.

A special class of ODEs is known as stiff ODEs, which arise in the areas of science and mathematics. It is also an essential tool for modeling many physical situations such as electrical circuits, mechanical system, chemical processes and so forth. A typical example of a stiff equation is the equation representing the rate of formation of free radicals in a complex chemical reaction [2]. Some other cases are the relaxation oscillation of the Van der Pol equation famous in applied mechanics and chemical kinetic rate equations involving photo-dissociation such as those describing the behavior of atmospheric pollutants [3].

As matter of fact, no universally accepted definition of stiffness exists [4]. The ODE is considered to be stiff if the solution is slowly changing and there are nearby solutions that are rapidly changing [5]. A stiff system contains components with slow and rapid decay rate, in which makes it difficult to solve it using explicit methods [6]. The simple equation of stiff problem, $y' = \lambda y,\ y(0) = \alpha,$ where the eigenvalue λ is a negative real number and the exact solution to this equation contains the transient solution $e^{\lambda x}$ or transient response of the system [7]. The stiff ODEs are challenging to solve since some of the numerical methods have absolute stability limitation on the step size [8].

The linear multistep methods are very famous for solving IVPs of ODEs but need single-step methods such as Euler's method and Runge-Kutta method for starting the solution where only the initial conditions are available [9]. They are called single-step method because they use only the information from the last step computed. Euler's method is the simplest one-step method and a classical way of solving IVPs. It is a basic explicit method for numerical integration of ODEs. Euler proposed his method back in 1768 and it was the first numerical method used for solving IVPs. The general formula for Euler's method is given by

$$y_{n+1} = y_n + hf(x_n, y_n). \qquad (2)$$

Although the Euler approach is simple, it is not exactly accurate due to the low accuracy and poor stability behavior [1]. Various modifications have been made by previous researchers on (2) that would improve the approximation of the numerical solutions. The IVPs were solved using improved modified Euler's method (IMEM):

$$y_{n+1} = y_n + hf\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} f\left(x_n, y_n + \frac{h}{2} f(x_n, y_n)\right)\right). \qquad (3)$$

The method (6) improved upon is the modified Euler's method (MEM) [10]:

$$y_{n+1} = y_n + hf\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} f(x_n, y_n)\right). \qquad (4)$$

In the paper by [11], the accuracy of Euler's method and MEM in terms of percentage local truncation error was discussed. The results of both methods when solving first order ODEs are not enough accurate except when very small step size is taken. Generally, the MEM is more accurate than Euler's method. Recently, a new version of Euler's method (NEM) with small modification to the MEM was proposed by [12]:

$$y_{n+1} = y_n + hf\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} f\left(x_n, y_n + \frac{h}{2} f(x_n, y_n + hf(x_n, y_n + hf(x_n, y_n)))\right)\right). \qquad (5)$$

Moreover, the MEM and IMEM were applied with the step sizes $h = 0.1$ and $h = 0.05$ for the numerical solutions of IVPs. Unlike MEM and IMEM, the NEM takes less time on average to perform the functions evaluation. Stability and consistency analysis were also presented, and the method has been proved to be consistent.

There are various methods found in the literature for finding stiff solutions. The most commonly multistep methods used are based on backward differentiation formulas (BDF), where the earliest research has been done by [2]. The BDF is suitable for solving stiff ODEs due to its stability region [13]. Although the classical BDF has been very popular in the past, it has its drawback in terms of computational cost. This is because it depends on multiple previous points to generate an approximate solution at one point consecutively. Therefore, many researchers have proposed block methods to help circumvent this shortcoming [14]–[19]. Block method is well-known with its ability to compute solutions at multiple points simultaneously, thus reducing the computational time [20]. The continuously increasing number of studies on block methods for solving ODEs have led to the competition in deriving an efficient algorithm for solving different types of ODEs [21]. Block methods can be formed either with one-step or multi-step methods. [22] stated that in one-step block methods, the approximation values obtained in each new block depend only on the last point of the previous block, while in multistep block methods, all points of the previous block are used to obtain the approximation values for the new block.

In pursuit of solving stiff problem, [22] has extended the classical BDF to develop block backward differentiation formulas (BBDF). There is no doubt that the work of [23] in particular fully implicit r-point BBDF has had a very beneficial impact on the numerical solution of stiff ODEs. In a related study, [24] studied the zero stability and consistency which are the necessary conditions for the convergence of the 2-point BBDF methods. It is proven that this method can solve stiff problems due to its A-stable properties. BBDF method is symmetrical to BDF method but the advantage of the BBDF method is its ability to produce more than one solution [25]. However, simple iterative scheme is not appropriate for solving stiff problems, especially when dealing with non-linear case. One way to overcome this issue is through the implementation of Newton's iteration. Nevertheless, the drawback is that it requires expensive task in Jacobian matrix evaluation per iteration.

Most of the previous researchers used the simplest Euler's method as starting method for the implementation of BBDF method. The main contribution in this study is to propose MEM, IMEM and NEM for finding starting values that give better approximation over the original BBDF approach. No detailed study of the implementation of modified versions of Euler's method as starting methods for BBDF has been done yet. It is sufficient to apply the methods proposed by [26] and [22] as to enhance the accuracy of BBDF in terms of absolute error, maximum error and average error.

2. Methodology

Before solving the IVPs, it is worth considering a revision to the research methodology. This section presents a brief review of the predictor-corrector BBDF method based on two derivation techniques.

2.1. Lagrange Interpolation Polynomial

The predictor-corrector formulas can be derived using interpolating polynomial, $P_k(x)$ which interpolates the values of y at the points x_{n-1}, x_n, x_{n+1} and x_{n+2}.

$$P_k(x)=\sum_{j=0}^{k} L_{k,j}(x) f\left(x_{n+1-j}\right), \tag{6}$$

where

$$L_{k,j}(x)=\prod_{\substack{i=0\\ i\neq j}}^{k}\frac{\left(x-x_{n+1-i}\right)}{\left(x_{n+1-j}-x_{n+1-i}\right)}$$

for each $j=0,\ 1,\ \ldots,\ k$. By introducing $x=x_{n+1}+sh$ and differentiating the resulting polynomial once with respect to s followed by substituting $s=0$ and $s=1$, the first point, y_{n+1} and second point, y_{n+2} of corrector formula will be obtained.

2.2. Backward Difference Interpolation Polynomial

Alternatively, the method can be derived using backward difference interpolation polynomial, $P_{k,n+1}(x)$ which interpolates $y(x)$ at k points:

$$P_{k,n+1}(x)=\sum_{m=0}^{k}(-1)^m\binom{-s}{m}\nabla^m y_{n+1}, \tag{7}$$

where $s=\dfrac{x-x_{n+1}}{h}$ and $m=0,1,2,\ldots$. By differentiating (7) once at $x=x_{n+1}$, the following equation is obtained:

$$P'_{k,n+1}(x)=\frac{1}{h}\sum_{m=0}^{k}\delta_{1,m}\nabla^m y_{n+1}, \tag{8}$$

where the coefficients $\delta_{1,m}$ can be produced using generating function, $D(x)=\sum_{m=0}^{\infty}\delta_{1,m}x^m$.

2.3. The Predictor-Corrector Formula

The 2-point BBDF method with constant step size proposed by [23] is employed as corrector formulas:

$$\left.\begin{aligned} y_{n+1}&=-\frac{1}{3}y_{n-1}+2y_n-\frac{2}{3}y_{n+2}+2hf_{n+1},\\ y_{n+2}&=\frac{2}{11}y_{n-1}-\frac{9}{11}y_n+\frac{18}{11}y_{n+1}+\frac{6}{11}hf_{n+2}.\end{aligned}\right\} \tag{9}$$

Formula (9) is fully implicit. Thus, the predictor is required to compute the starting values, y_{n-1} and y_n as well as the future values, y_{n+2} and f_{n+1}. The predictor formula can be derived in the similar manner without differentiation process. The predictor formula for (9) is given by

$$\left.\begin{aligned} y_{n+1}&=y_{n-2}-3y_{n-1}+3y_n,\\ y_{n+2}&=3y_{n-2}-8y_{n-1}+6y_n.\end{aligned}\right\} \tag{10}$$

The solutions of y_{n+1} and y_{n+2} at the points, x_{n+1} and x_{n+2} will be approximated simultaneously using two previous values, y_{n-1} and y_n at the points, x_{n-1} and x_n .

2.4. Implementation of Newton's Iteration

Generally, fixed point iteration is inappropriate for stiff problems. Therefore, the iteration scheme based on Newton's iteration method is applied into (9), which can be rewritten in the following form:

$$\left.\begin{aligned} F_1&=y_{n+1}+\frac{2}{3}y_{n+2}-2hf_{n+1}-\varsigma_1=0,\\ F_2&=-\frac{18}{11}y_{n+1}+y_{n+2}-\frac{6}{11}hf_{n+2}-\varsigma_2=0,\end{aligned}\right\} \tag{11}$$

where ς_1 and ς_2 are the previous values.

The difference between i and $(i+1)^{th}$ iterations for y_{n+1} and y_{n+2} are given as follows:

$$e_{n+1}^{(i+1)}=y_{n+1}^{(i+1)}-y_{n+1}^{(i)} \tag{12}$$

$$e_{n+2}^{(i+1)}=y_{n+2}^{(i+1)}-y_{n+2}^{(i)} \tag{13}$$

By applying the Newton's method to (11), yields

$$y_{n+1}^{(i+1)}=y_{n+1}^{(i)}-\frac{F_1(y_{n+1}^{(i)})}{F_1'(y_{n+1}^{(i)})}, \tag{14}$$

$$y_{n+2}^{(i+1)}=y_{n+2}^{(i)}-\frac{F_2(y_{n+2}^{(i)})}{F_2'(y_{n+2}^{(i)})}, \tag{15}$$

where $y_{n+1}^{(i+1)}$ and $y_{n+2}^{(i+1)}$ are the $(i+1)^{th}$ iterative value of

y_{n+1} and y_{n+2} respectively. The iteration is generated until $\left|y_{n+1}^{(i+1)}-y_{n+1}^{(i)}\right|$ and $\left|y_{n+2}^{(i+1)}-y_{n+2}^{(i)}\right|$ are sufficiently small. Hence, we have

$$y_{n+1}^{(i+1)}-y_{n+1}^{(i)}=\frac{-\left(y_{n+1}^{(i)}+\frac{2}{3}y_{n+2}^{(i)}-2hf_{n+1}^{(i)}-\varsigma_1\right)}{1-2h\frac{\partial f_{n+1}}{\partial y_{n+1}}}. \quad (16)$$

It follows that

$$\left(1-2h\frac{\partial f_{n+1}}{\partial y_{n+1}}\right)\left(y_{n+1}^{(i+1)}-y_{n+1}^{(i)}\right)= =-y_{n+1}^{(i)}-\frac{2}{3}y_{n+2}^{(i)}+2hf_{n+1}^{(i)}+\varsigma_1. \quad (17)$$

Similarly, the Newton's iteration for the second point, y_{n+2} takes the form

$$y_{n+2}^{(i+1)}-y_{n+2}^{(i)}=\frac{-\left(-\frac{18}{11}y_{n+1}^{(i)}+y_{n+2}^{(i)}-\frac{6}{11}hf_{n+2}^{(i)}-\varsigma_2\right)}{1-\frac{6}{11}h\frac{\partial f_{n+2}}{\partial y_{n+2}}}. \quad (18)$$

It follows that

$$\left(1-\frac{6}{11}h\frac{\partial f_{n+2}}{\partial y_{n+2}}\right)\left(y_{n+2}^{(i+1)}-y_{n+2}^{(i)}\right)= =\frac{18}{11}y_{n+1}^{(i)}-y_{n+2}^{(i)}+\frac{6}{11}hf_{n+2}^{(i)}+\varsigma_2. \quad (19)$$

The resulting equations (17) and (19) are transformed into the matrix form:

$$\underbrace{\begin{bmatrix}1-2h\left(\frac{\partial f_{n+1}}{\partial y_{n+1}}\right) & \frac{2}{3}\\ -\frac{18}{11} & 1-\frac{6}{11}h\left(\frac{\partial f_{n+2}}{\partial y_{n+2}}\right)\end{bmatrix}}_{\text{Jacobian matrix}}\begin{bmatrix}e_{n+1}^{(i+1)}\\ e_{n+2}^{(i+1)}\end{bmatrix} =\begin{bmatrix}-1 & -\frac{2}{3}\\ \frac{18}{11} & -1\end{bmatrix}\begin{bmatrix}y_{n+1}^{(i)}\\ y_{n+2}^{(i)}\end{bmatrix}+h\begin{bmatrix}2 & 0\\ 0 & \frac{6}{11}\end{bmatrix}\begin{bmatrix}f_{n+1}^{(i)}\\ f_{n+2}^{(i)}\end{bmatrix}+\begin{bmatrix}\varsigma_1\\ \varsigma_2\end{bmatrix}. \quad (20)$$

The corresponding linear system to be solved is $AE_{1,2}^{(i+1)}=B$, where

$$A=\begin{bmatrix}1-2h\left(\frac{\partial f_{n+1}}{\partial y_{n+1}}\right) & \frac{2}{3}\\ -\frac{18}{11} & 1-\frac{6}{11}h\left(\frac{\partial f_{n+2}}{\partial y_{n+2}}\right)\end{bmatrix},$$

and

$$B=\begin{bmatrix}-y_{n+1}^{(i)}-\frac{2}{3}y_{n+2}^{(i)}+2hf_{n+1}^{(i)}+\varsigma_1\\ \frac{18}{11}y_{n+1}^{(i)}-y_{n+2}^{(i)}+\frac{6}{11}hf_{n+2}^{(i)}+\varsigma_2\end{bmatrix},$$

while $E_{1,2}^{(i+1)}=\begin{bmatrix}e_{n+1}^{(i+1)}\\ e_{n+2}^{(i+1)}\end{bmatrix}$ is the increment defined as (12) and (13). By using LU decomposition, $AE_{1,2}^{(i+1)}=B$ is solved to obtain the approximation values of $y_{n+1}^{(i+1)}$ and $y_{n+2}^{(i+1)}$ simultaneously. In the early stage of iteration, the modified versions of Euler's methods (3)-(5) are used to find the starting values over sub-interval $\left[x_{n-1},x_n\right]$. Then two-stage Newton's iteration, where $i=0,1$ is performed in predictor-evaluate-corrector-evaluate (PECE) scheme:

For $i=0$:

P: Estimation of predicted values, $y_{n+1}^{(i)}$ and $y_{n+2}^{(i)}$.

E: Evaluation of the functions, $f_{n+1}^{(i)}$ and $f_{n+2}^{(i)}$.

For $i=0,1$:

C: Computation of corrected values, $y_{n+1}^{(i+1)}$ and $y_{n+2}^{(i+1)}$ with $e_{n+1}^{(i+1)}$ and $e_{n+2}^{(i+1)}$, where $i=0,1$.

E: Evaluation of the functions, $f_{n+1}^{(i+1)}$ and $f_{n+2}^{(i+1)}$.

3. Numerical Results

Four numerical examples of linear and non-linear stiff IVPs with transient and steady-state solutions are selected to examine the applicability of the methods. The graphs of exact and approximate solutions for each tested problem are illustrated in Figure 1-4. A central processing unit (CPU) time is used to quantify the empirical efficiency of the algorithm. Table 1-4 present the absolute errors, while Table 5-8 present the numerical results of maximum error (MAXE), average error (AVER) and computational time (TIME).

The absolute error is defined as follows:

$$\text{error}_i^T=\left|y_{i,\text{ approximate}}^T-y_{i,\text{ exact}}^T\right|.$$

Hence, the formulas of MAXE and AVER are given as follows:

$$\text{MAXE}=\max_{1\le i\le T}\left(\max_{1\le i\le N}(\text{error}_i^T)\right),$$

$$\text{AVER}=\frac{\sum_{i=1}^{T}\sum_{i=1}^{N}(\text{error}_i^T)}{T},$$

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

Because a constant step size is employed, a value of step size h must be chosen prior to the computation. To show the effects of stiffness and obtain better approximation, h is gradually decreased. The following notations are used in Table 1-8:

- h: Step size
- BBDF: BBDF given by [23]
- BBDF-MEM: 2-point BBDF with MEM
- BBDF-IMEM: 2-point BBDF with IMEM
- BBDF-NEM: 2-point BBDF with NEM

- AVGE: Average error
- MAXE: Maximum error
- TIME: CPU time (second)

Problem 1:

Linear problem

$$y' = -y, \quad y(0) = 1, \quad 0 \le x \le 1.$$

Exact solution: $y(x) = e^{-x}$.
Source: [22]

Problem 2:

Linear problem

$$y' = -10y + 10, \quad y(0) = 2, \quad 0 \le x \le 1.$$

Exact solution: $y(x) = 1 + e^{-10x}$.
Source: [22]

Problem 3:

Non-linear problem

$$y' = \frac{y(1-y)}{2y-1}, \quad y(0) = \frac{5}{6}, \quad 0 \le x \le 1.$$

Exact solution: $y(x) = \frac{1}{2} + \sqrt{\frac{1}{4} - \frac{5}{36}e^{-x}}$.

Source: [27]

Problem 4:

Non-linear problem

$$y' = \frac{50}{y} - 50y, \quad y(0) = \sqrt{2}, \quad 0 \le x \le 1.$$

Exact solution: $y(x) = \sqrt{1 + e^{-100x}}$.
Source: [27]

Table 1. Absolute errors for Problem 1

h	*x*	BBDF	BBDF-MEM	BBDF-IMEM	BBDF-NEM
0.1	0.1	4.83742e-3	1.62582e-4	8.74180e-5	6.49180e-5
	0.2	8.73075e-3	2.94247e-4	1.58191e-4	1.17476e-4
	0.3	1.18182e-2	3.99404e-4	2.14695e-4	1.59440e-4
	0.4	1.15700e-2	3.83065e-4	2.17977e-4	1.63898e-4
	0.5	1.02807e-2	3.44115e-4	1.90088e-4	1.42022e-4
	0.6	1.81913e-2	8.86276e-3	9.33175e-3	9.28956e-3
	0.7	2.75020e-2	1.95117e-2	1.99133e-2	1.98772e-2
	0.8	3.62777e-2	2.93478e-2	2.96961e-2	2.96648e-2
	0.9	4.38166e-2	3.77992e-2	3.81017e-2	3.80745e-2
	1.0	4.92915e-2	4.40317e-2	4.42961e-2	4.42723e-2
0.05	0.1	2.33742e-3	3.91445e-5	2.03077e-5	1.74838e-5
	0.2	3.41454e-3	5.66015e-5	3.02964e-5	2.61689e-5
	0.3	6.00375e-3	2.94601e-3	3.02256e-3	3.01892e-3
	0.4	1.29142e-2	1.02193e-2	1.02868e-2	1.02836e-2
	0.5	1.87808e-2	1.63750e-2	1.64352e-2	1.64324e-2
	0.6	2.32547e-2	2.11028e-2	2.11567e-2	2.11541e-2
	0.7	2.66179e-2	2.46933e-2	2.47415e-2	2.47392e-2
	0.8	2.90705e-2	2.73495e-2	2.73926e-2	2.73905e-2
	0.9	3.07629e-2	2.92239e-2	2.92625e-2	2.92606e-2
	1.0	3.18227e-2	3.04466e-2	3.04810e-2	3.04794e-2
0.01	0.1	1.03892e-3	8.94726e-4	8.95447e-4	8.95439e-4
	0.2	2.62461e-3	2.49440e-3	2.49505e-3	2.49505e-3
	0.3	3.89598e-3	3.77841e-3	3.77900e-3	3.77899e-3
	0.4	4.89878e-3	4.79262e-3	4.79315e-3	4.79314e-3
	0.5	5.67288e-3	5.57702e-3	5.57750e-3	5.57749e-3
	0.6	6.25297e-3	6.16641e-3	6.16684e-3	6.16684e-3
	0.7	6.66920e-3	6.59104e-3	6.59143e-3	6.59142e-3
	0.8	6.94769e-3	6.87712e-3	6.87747e-3	6.87747e-3
	0.9	7.11109e-3	7.04736e-3	7.04768e-3	7.04767e-3
	1.0	7.17893e-3	7.12138e-3	7.12167e-3	7.12167e-3

Table 2. Absolute errors for Problem 2

h	*x*	**BBDF**	**BBDF-MEM**	**BBDF-IMEM**	**BBDF-NEM**
0.1	0.1	3.67879e-1	1.32121e-1	1.17879e-1	1.17879e-1
	0.2	1.35335e-1	1.14665e-1	7.28353e-2	7.28353e-2
	0.3	4.97871e-2	7.52129e-2	3.41621e-2	3.41621e-2
	0.4	1.83156e-2	3.32717e-2	1.53394e-2	1.53394e-2
	0.5	6.73795e-3	1.11192e-2	4.50580e-3	4.50580e-3
	0.6	2.47875e-3	1.51394e-2	2.05640e-3	2.05640e-3
	0.7	9.11882e-4	5.05352e-4	1.51383e-2	1.51383e-2
	0.8	3.35463e-4	3.09517e-2	2.35348e-2	2.35348e-2
	0.9	1.23410e-4	8.79205e-2	3.50695e-2	3.50695e-2
	1.0	4.53999e-5	1.15822e-1	3.16537e-2	3.16537e-2
0.05	0.1	1.17879e-1	2.27456e-2	1.53404e-2	6.00200e-3
	0.2	6.65853e-2	1.35905e-2	1.02941e-2	4.57879e-3
	0.3	3.16621e-2	2.31757e-2	2.53848e-2	2.48373e-2
	0.4	6.69064e-3	3.77004e-2	2.73384e-2	2.97515e-2
	0.5	1.70058e-2	2.51676e-3	4.48987e-3	2.88361e-3
	0.6	2.45156e-2	3.80024e-2	3.44385e-2	3.53181e-2
	0.7	1.33284e-2	4.24578e-2	3.34642e-2	3.55977e-2
	0.8	2.70127e-3	1.35987e-2	8.26220e-3	9.51077e-3
	0.9	9.05123e-3	1.54547e-2	1.36482e-2	1.40864e-2
	1.0	3.65327e-3	1.97981e-2	1.46874e-2	1.58926e-2
0.01	0.1	4.92915e-2	4.40317e-2	4.42961e-2	4.42723e-2
	0.2	4.76683e-2	4.62183e-2	4.62912e-2	4.62846e-2
	0.3	2.56134e-2	2.52136e-2	2.52336e-2	2.52318e-2
	0.4	1.16499e-2	1.15396e-2	1.15452e-2	1.15447e-2
	0.5	4.89989e-3	4.86949e-3	4.87102e-3	4.87088e-3
	0.6	1.97192e-3	1.96353e-3	1.96396e-3	1.96392e-3
	0.7	7.72125e-4	7.69813e-4	7.69929e-4	7.69919e-4
	0.8	2.96925e-4	2.96288e-4	2.96320e-4	2.96317e-4
	0.9	1.12783e-4	1.12608e-4	1.12616e-4	1.12616e-4
	1.0	4.24697e-5	4.24213e-5	4.24237e-5	4.24235e-5

Table 3. Absolute errors for Problem 3

h	*x*	**BBDF**	**BBDF-MEM**	**BBDF-IMEM**	**BBDF-NEM**
0.1	0.1	1.56471e-3	9.24821e-5	3.68131e-5	1.99789e-5
	0.2	2.58122e-3	1.47941e-4	6.00729e-5	3.36664e-5
	0.3	3.24199e-3	1.80910e-4	7.47334e-5	4.29633e-5
	0.4	3.07236e-3	1.62691e-4	7.79934e-5	4.81901e-5
	0.5	2.65455e-3	1.44250e-4	6.40207e-5	3.82085e-5
	0.6	4.17877e-3	1.82506e-3	2.00017e-3	1.97845e-3
	0.7	5.87580e-3	3.96285e-3	4.10524e-3	4.08754e-3
	0.8	7.45102e-3	5.87394e-3	5.99131e-3	5.97670e-3
	0.9	8.71737e-3	7.41458e-3	7.51151e-3	7.49943e-3
	1.0	9.53504e-3	8.44016e-3	8.52159e-3	8.51143e-3
0.05	0.1	7.36116e-4	2.17027e-5	7.96245e-6	5.84222e-6
	0.2	1.00676e-3	2.80291e-5	1.17900e-5	8.99194e-6
	0.3	1.60679e-3	7.36955e-4	7.70442e-4	7.68087e-4
	0.4	3.20826e-3	2.48444e-3	2.51232e-3	2.51036e-3
	0.5	4.45105e-3	3.83343e-3	3.85721e-3	3.85554e-3
	0.6	5.26080e-3	4.72711e-3	4.74766e-3	4.74621e-3
	0.7	5.76941e-3	5.30548e-3	5.32333e-3	5.32208e-3
	0.8	6.06708e-3	5.66209e-3	5.67768e-3	5.67658e-3
	0.9	6.20972e-3	5.85497e-3	5.86862e-3	5.86766e-3
	1.0	6.23668e-3	5.92503e-3	5.93702e-3	5.93618e-3
0.01	0.1	3.20618e-4	2.74894e-4	2.75261e-4	2.75255e-4
	0.2	7.47203e-4	7.07929e-4	7.08244e-4	7.08239e-4
	0.3	1.03526e-3	1.00123e-3	1.00150e-3	1.00150e-3
	0.4	1.22726e-3	1.19757e-3	1.19781e-3	1.19780e-3
	0.5	1.35072e-3	1.32468e-3	1.32489e-3	1.32489e-3
	0.6	1.42433e-3	1.40140e-3	1.40158e-3	1.40158e-3
	0.7	1.46121e-3	1.44094e-3	1.44110e-3	1.44110e-3
	0.8	1.47081e-3	1.45285e-3	1.45300e-3	1.45299e-3
	0.9	1.46014e-3	1.44418e-3	1.44431e-3	1.44431e-3
	1.0	1.43442e-3	1.42022e-3	1.42033e-3	1.42033e-3

Table 4. Absolute errors for Problem 4

h	*x*	BBDF	BBDF-MEM	BBDF-IMEM	BBDF-NEM
0.1	0.1	3.12134e+0	1.19602e+1	2.32315e+1	2.87096e+2
	0.2	5.12826e+0	9.27124e+1	5.50589e+2	1.85315e+5
	0.3	2.46971e+1	7.80383e+2	1.25021e+4	1.20038e+8
	0.4	5.72093e+0	1.46163e+2	2.39320e+3	2.28211e+7
	0.5	2.42068e+0	1.01880e+2	1.72127e+3	1.63399e+7
	0.6	5.12790e+0	1.49192e+2	2.80563e+3	2.59311e+7
	0.7	1.09458e+1	4.49436e+2	6.39936e+3	6.34476e+7
	0.8	4.42161e+1	1.58787e+3	2.45918e+4	2.38288e+8
	0.9	8.23134e+1	2.88588e+3	4.53112e+4	4.37471e+8
	1.0	8.29092e+1	2.80523e+3	4.46299e+4	4.29386e+8
0.05	0.1	7.54076e+0	3.49456e+0	7.85398e+0	3.26778e+2
	0.2	2.67472e+0	4.07137e-1	7.76432e+0	1.92245e+3
	0.3	7.84205e+0	6.47702e-1	1.15333e+1	3.29771e+3
	0.4	3.98991e+0	7.79802e+0	2.37500e-1	1.37351e+3
	0.5	1.31143e+1	1.80153e+1	3.39319e+1	1.17472e+4
	0.6	1.35164e+1	4.99045e+1	3.01717e+1	8.75653e+3
	0.7	1.00352e+1	1.74590e+2	1.61255e+1	5.46434e+3
	0.8	1.36136e+1	5.77137e+2	1.38116e+1	1.31579e+3
	0.9	2.44801e+1	1.86713e+3	6.61960e+1	2.48545e+4
	1.0	4.45598e+1	6.06480e+3	9.27636e+1	2.90925e+4
0.01	0.1	3.16697e-3	3.77646e-2	1.32185e-2	1.50935e-2
	0.2	1.37012e-3	3.57246e-2	1.49231e-2	1.66842e-2
	0.3	5.85637e-4	4.30626e-2	1.99795e-2	2.21158e-2
	0.4	3.57514e-4	5.59950e-2	2.76575e-2	3.04187e-2
	0.5	1.97897e-3	7.43288e-2	3.84904e-2	4.21037e-2
	0.6	4.79900e-3	9.93899e-2	5.36070e-2	5.83479e-2
	0.7	9.57225e-3	1.33613e-1	7.47053e-2	8.09392e-2
	0.8	1.74903e-2	1.80840e-1	1.04274e-1	1.12510e-1
	0.9	3.04945e-2	2.47258e-1	1.46063e-1	1.57060e-1
	1.0	5.18609e-2	3.43610e-1	2.06074e-1	2.21082e-1

Table 5. Numerical results of for Problem 1

h	**Method**	**MAXE**	**AVGE**	**TIME**
0.1	BBDF	4.38166e-2	1.47638e-2	0.000790
	BBDF-MEM	3.77992e-2	9.62486e-3	0.000557
	BBDF-IMEM	3.81017e-2	9.74509e-3	0.000546
	BBDF-NEM	3.80745e-2	9.72119e-3	0.000507
0.05	BBDF	3.13617e-2	1.57570e-2	0.000900
	BBDF-MEM	2.99064e-2	1.39262e-2	0.001089
	BBDF-IMEM	2.99428e-2	1.39613e-2	0.000988
	BBDF-NEM	2.99411e-2	1.39591e-2	0.000877
0.01	BBDF	7.17594e-3	4.82818e-3	0.060709
	BBDF-MEM	7.11780e-3	4.73386e-3	0.037115
	BBDF-IMEM	7.11810e-3	4.73431e-3	0.032898
	BBDF-NEM	7.11809e-3	4.73431e-3	0.030484

Table 6. Numerical results for Problem 2

h	**Method**	**MAXE**	**AVGE**	**TIME**
0.1	BBDF	1.83156e-2	2.89031e-3	0.000670
	BBDF-MEM	8.79205e-2	1.78908e-2	0.000427
	BBDF-IMEM	3.50695e-2	9.56443e-3	0.000429
	BBDF-NEM	3.50695e-2	9.56443e-3	0.000449
0.05	BBDF	6.65853e-2	1.52225e-2	0.001219
	BBDF-MEM	4.95115e-2	1.91390e-2	0.001038
	BBDF-IMEM	4.14038e-2	1.57808e-2	0.000891
	BBDF-NEM	4.33446e-2	1.59949e-2	0.000794
0.01	BBDF	5.67155e-2	1.38152e-2	0.038563
	BBDF-MEM	5.35777e-2	1.28665e-2	0.034273
	BBDF-IMEM	5.37354e-2	1.29004e-2	0.028694
	BBDF-NEM	5.37212e-2	1.28960e-2	0.032656

Table 7. Numerical results for Problem 3

h	**Method**	**MAXE**	**AVGE**	**TIME**
0.1	BBDF	8.71737e-3	3.19499e-3	0.000929
	BBDF-MEM	7.41458e-3	1.93834e-3	0.000624
	BBDF-IMEM	7.51151e-3	1.97502e-3	0.000693
	BBDF-NEM	7.49943e-3	1.96285e-3	0.000445
0.05	BBDF	6.23457e-3	3.48023e-3	0.001021
	BBDF-MEM	5.90215e-3	3.00441e-3	0.000799
	BBDF-IMEM	5.91494e-3	3.01751e-3	0.000844
	BBDF-NEM	5.91404e-3	3.01621e-3	0.000754
0.01	BBDF	1.47086e-3	1.11270e-3	0.032888
	BBDF-MEM	1.45285e-3	1.08607e-3	0.029328
	BBDF-IMEM	1.45300e-3	1.08627e-3	0.031933
	BBDF-NEM	1.45299e-3	1.08627e-3	0.029967

Table 8. Numerical results for Problem 4

h	**Method**	**MAXE**	**AVGE**	**TIME**
0.1	BBDF	8.23134e+1	1.50745e+1	0.000584
	BBDF-MEM	2.88588e+3	5.32042e+2	0.000452
	BBDF-IMEM	4.53112e+4	8.32225e+3	0.000474
	BBDF-NEM	4.37471e+8	8.04298e+7	0.000424
0.05	BBDF	4.67972e+1	9.82069e+0	0.001231
	BBDF-MEM	4.37496e+3	4.50708e+2	0.001043
	BBDF-IMEM	1.08886e+2	2.14701e+1	0.000939
	BBDF-NEM	3.76349e+4	7.23961e+3	0.000938
0.01	BBDF	1.44729e-1	2.95950e-2	0.035103
	BBDF-MEM	3.11941e-1	7.59236e-2	0.033325
	BBDF-IMEM	1.76967e-1	3.77501e-2	0.028258
	BBDF-NEM	1.96718e-1	4.08597e-2	0.027080

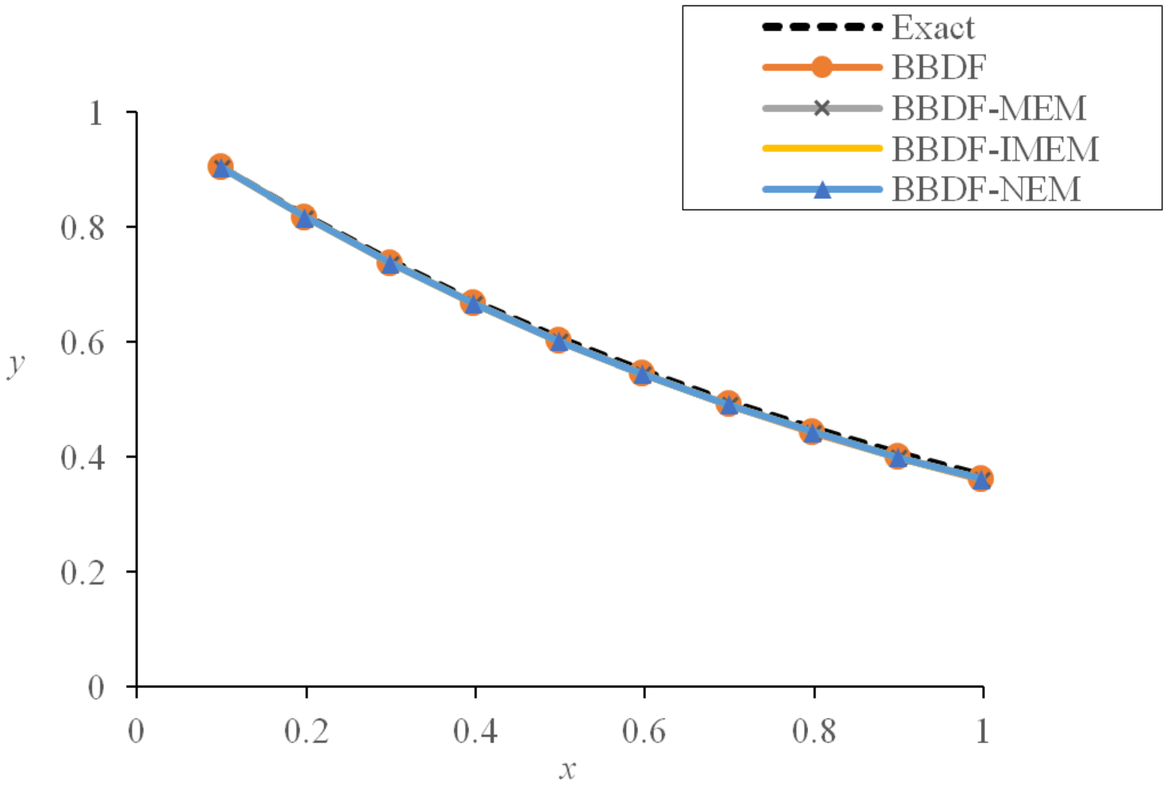

Figure 1. Graph of exact and approximate solutions for Problem 1 at $h = 0.01$

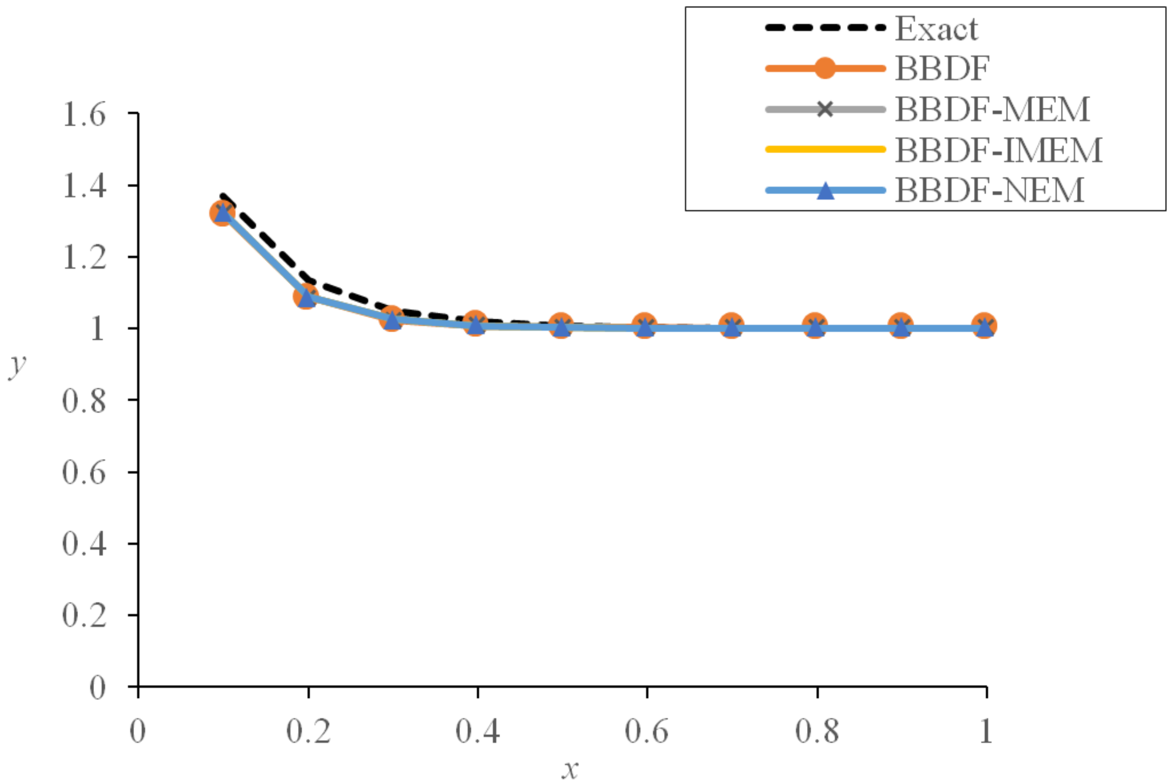

Figure 2. Graph of exact and approximate solutions for Problem 2 at $h = 0.01$

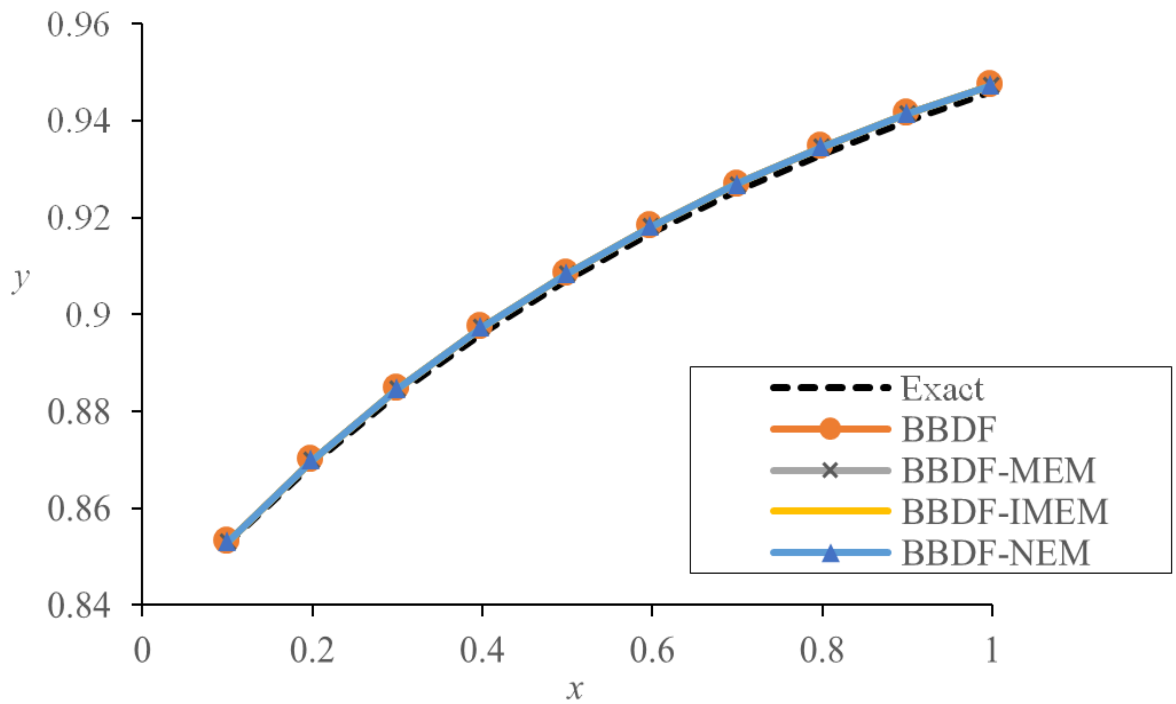

Figure 3. Graph of exact and approximate solutions for Problem 3 at $h = 0.01$

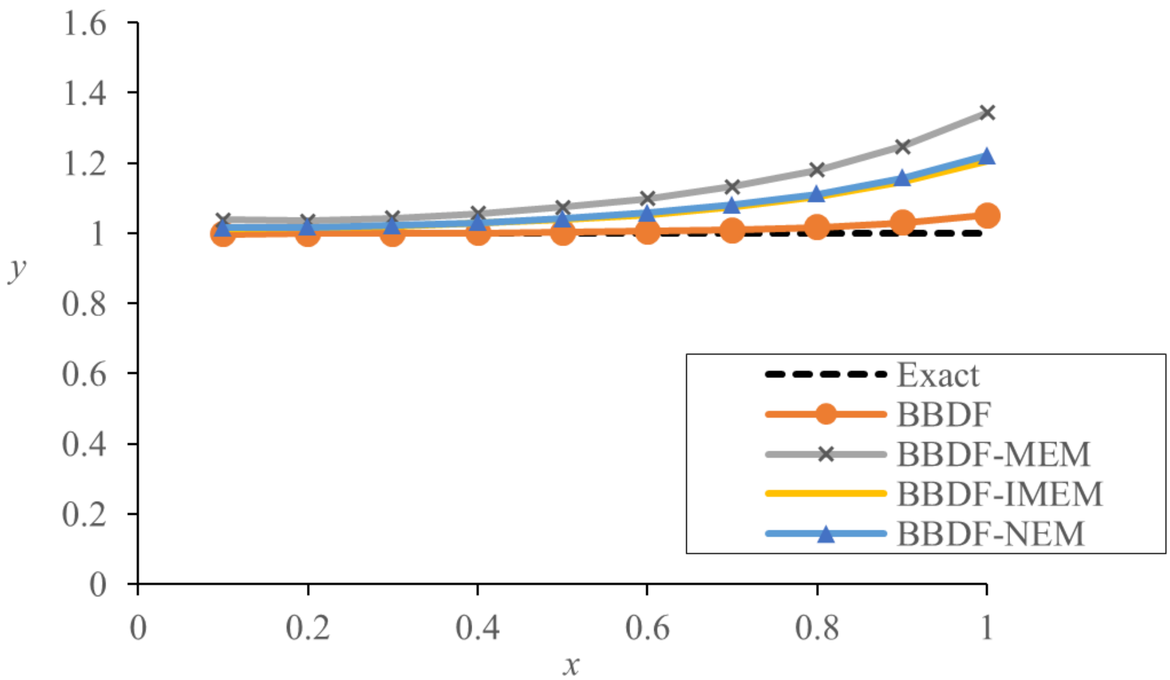

Figure 4. Graph of exact and approximate solutions for Problem 4 at $h = 0.01$

4. Discussion

As in Table 1, the BBDF-MEM, BBDF-IMEM, and BBDF-NEM provide a much better approximation for Problem 1. It is observed that the absolute errors produced by the proposed methods are smaller than the BBDF for most of the step size and point of approximation. In the case where $h = 0.1$ and $h = 0.05$, it is clear from Table 2 that the absolute errors obtained by the proposed methods are smaller than that in the BBDF in the early points. It is significant that the proposed methods give smallest absolute error at early points with $h = 0.1$. However, in the case where $h = 0.01$, the results are comparable. As presented in Table 3, the proposed methods give the smallest absolute error at certain points and step size. Inspection of Table 4 shows that all methods disagree with the analytical solutions. It is seen that all methods have a large absolute error with $h = 0.1$ and $h = 0.05$. The exact solution of Problem 4 shows the large negative λ, which means very stiff. The step size required to maintain stability of the method might be much smaller than the step size required to solve such problem precisely. In light of these observations, it is becoming increasingly apparent that the methods manage to achieve desired accuracy when the step size decreases to $h = 0.01$.

As shown in Table 5, the proposed methods slightly produce smaller maximum error and average error than the BBDF. From Table 6, in the case where $h = 0.1$, the BBDF-MEM has larger maximum error and average error compared to the other methods. However, the proposed methods prevail the BBDF when the step size is getting smaller. From Table 7, there is no significant difference on maximum error as well as the average error for all methods. Not surprisingly, Table 8 shows large maximum error and absolute error when $h = 0.1$ and $h = 0.05$, which leads to the fact that the methods applied are instable when the step size used are not small enough for very stiff equations.

On comparing the nature of approximate solution with the analytical solutions available in Problem 1-4, the graphical representations for the case of $h = 0.01$ are plotted in Figures 1-4, which exhibit that the results obtained by the numerical methods are competent with exact solution. In Figure 1, the solution curves decrease gradually as x increases. In Figure 2, the solution curves decrease rapidly to 1 as x increases. Also in Figure 3, the solution curves are gradually increasing with the increase of x. However, the solution trend shows that BBDF and exact solution are overlapping to each other, whereas there is a little difference between exact solution and that of proposed methods as depicted in Figure 4.

The efficiency of the method is evaluated by CPU time required to complete the computation. Table 5-8 indicate that the overall computational time of the BBDF is improved by adopting the modified versions of Euler's methods compared to the conventional Euler's method. The BBDF-NEM takes less time than BBDF-IMEM and BBDF-MEM. This can be justified by the fact that the NEM is the fastest method to perform functions evaluation followed by MEM and IMEM [26].

5. Conclusions

The purpose of this work is to propose a new implementation of 2-point BBDF by employing MEM, IMEM and NEM as starting methods for solving stiff IVPs of first order ODEs. While the accuracy of the methods can be improved by taking larger step size, the proposed methods avail competent results with smaller step size. The reduction in absolute error and average error at certain step size and stage of iteration is further evidence of the significance of modified versions of Euler's method in improving the accuracy of BBDF. The general conclusion to emerge from this sort of approach is that, even though the application of these new versions of starting method does not promise the best implementation, the improvement of the accuracy and execution time of multistep method in solving stiff problems is worth the effort.

Acknowledgments

This research was supported by Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA (UiTM) Shah Alam and Ministry of Higher Education (MoHE) Malaysia. The publication of this paper was funded by Geran Penyelidikan Khas UiTM 2020 (600-RMC/GPK 5/3 (272/2020)).

REFERENC ES

[1] S. L. Ross, "Introduction to Ordinary Differential Equations," *Applied Engineering Mathematics*. John Wiley & Sons, pp. 119–138, 2020. doi: 10.1201/9781003002260-7.

[2] C. F. Curtiss and J. O. Hirschfelder, "Integration of Stiff Equations," *Proceedings of the National Academy of Sciences*, vol. 38, no. 3, pp. 235–243, 1952, doi: 10.1073/pnas.38.3.235.

[3] L. F. Shampine and C. W. Gear, "A User's View of Solving Stiff Ordinary Differential Equations," *SIAM Review*, vol. 21, no. 1, pp. 1–17, 1979, doi: 10.1137/1021001.

[4] L. Brugnano, F. Mazzia, and D. Trigiante, "Fifty years of stiffness," *Recent Advances in Computational and Applied Mathematics*, pp. 1–21, 2011, doi: 10.1007/978-90-481-99 81-5_1.

[5] Z. Jackiewicz, *General linear methods for ordinary differential equations*. 2009. doi: 10.1016/j.matcom.2007.0 2.006.

[6] H. Musa, M. Bin Suleiman, F. Ismail, N. Senu, Z. A. Majid, and Z. B. Ibrahim, "A new fifth order implicit block method for solving first order stiff ordinary differential equations," *Malaysian Journal of Mathematical Sciences*, vol. 8, pp. 45–59, 2014.

[7] I. Asnor, S. A. M. Yatim, and Z. B. Ibrahim, "Modified higher order block backward differentiation formula for third order ordinary differential equations," *Journal of Physics: Conference Series*, vol. 1988, no. 1, 2021, doi: 10.1088/1742-6596/1988/1/012004.

[8] N. Mohamad Noor, Z. B. Ibrahim, and F. Ismail, "Numerical solution for stiff initial value problems using 2-point block multistep method," *Journal of Physics: Conference Series*, vol. 1132, no. 1, 2018, doi: 10.1088/1742-6596/1132/1/012017.

[9] L. Mulatu, A. Shiferaw, and S. Gebregiorgis, "Block procedure for solving stiff initial value problems using probabilists Hermite polynomials," *Engineering and Applied Science Letters*, vol. 3, no. 3, pp. 20–29, 2020, doi: 10.30538/psrp-easl2020.0044.

[10] A. Ochoche, "Improving the Improved Modified Euler Method for Better Performance on Autonomous Initial Value Problems," no. 12, pp. 57–66, 2008.

[11] B.M.Hamed, I. Yuosif. I. A. Alrhaman, and I. Sani, "The Accuracy of Euler and modified Euler Technique for First Order Ordinary Differential Equations with initial condition," *American Journal of Engineering Research* (AJER), no. 6, pp. 334–338, 2017.

[12] H. Workie, "Small Modification on Modified Euler Method for Solving Initial Value Problems," *Abstract and Applied Analysis*, vol. 2021, pp. 0–4, 2021, doi: 10.1155/2021/9951815.

[13] W. Gear, *Numerical initial value problems in ordinary differential equations.* Englewood Cliffs, New Jersey.: Prentice-Hall, 1971.

[14] N. Abasi, M. Suleiman, Z. B. Ibrahim, and F. Ismail, "2-point and 3-Point BBDF methods for solving semi-explicit index-1 DAEs," *Applied Mathematical Sciences*, vol. 6, no. 133–136, pp. 6679–6689, 2012.

[15] N. A. Rasid, Z. B. Ibrahim, Z. A. Majid, and F. Ismail, "Formulation of a new implicit method for group implicit BBDF in solving related stiff ordinary differential equations," *Mathematics and Statistics*, vol. 9, no. 2, pp. 144–150, 2021, doi: 10.13189/ms.2021.090208.

[16] 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, doi: 10.5556/j.tkjm.51.2020.2964.

[17] N. A. A. M. Nasir, Z. B. Ibrahim, K. I. Othman, and M. Suleiman, "Numerical solution of first order stiff Ordinary Differential Equations using fifth order block backward differentiation formulas," *Sains Malaysiana*, vol. 41, no. 4, pp. 489–492, 2012.

[18] M. Bin Suleiman, H. Musa, and F. Ismail, "An implicit 2-point block extended backward differentiation formula for integration of stiff initial value problems," *Malaysian Journal of Mathematical Sciences*, vol. 9, no. 1, pp. 33–51, 2015.

[19] 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, 2019, doi: 10.3390/MATH7020211.

[20] A. A. Nasarudin, Z. B. Ibrahim, and H. Rosali, "On the Integration of Stiff ODEs Using Block Backward Differentiation Formulas of Order Six," *Symmetry*, vol. 12,

no. 952, pp. 1–13, 2020.

[21] S. M. Zawawi, Z. B. Ibrahim, F. Ismail, and Z. A. Majid, “Diagonally implicit block backward differentiation formulas for solving ordinary differential equations,” *International Journal of Mathematics and Mathematical Sciences*, vol. 2012, 2012, doi: 10.1155/2012/767328.

[22] Z. B. Ibrahim, “Block multistep methods for solving ordinary differential equations,” Universiti Putra Malaysia, 2006.

[23] Z. B. Ibrahim, K. I. Othman, and M. 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, doi: 10.1016/j.amc.2006.07.116.

[24] Z. B. Ibrahim, M. Suleiman, N. A. A. M. Nasir, and K. I. Othman, “Convergence of the 2-point block backward differentiation formulas,” *Applied Mathematical Sciences*, vol. 5, no. 69–72, pp. 3473–3480, 2011.

[25] A. I. Asnor, S. A. M. Yatim, and Z. B. Ibrahim, “Convergence analysis of modified variable step block backward differentiation formulae,” *International Journal of Engineering Research and Technology*, vol. 11, no. 9, pp. 1437–1449, 2018.

[26] H. Workie, “Small Modification on Modified Euler Method for Solving Initial Value Problems,” *Abstract and Applied Analysis*, vol. 2021, pp. 0–4, 2021, doi: 10.1155/2021/9951815.

[27] H. Musa, M. B. Suleiman, and N. Senu, “Fully implicit 3-point block extended backward differentiation formula for stiff initial value problems,” *Applied Mathematical Sciences*, vol. 6, no. 85–88, pp. 4211–4228, 2012.