Modified higher order block backward differentiation formula for third order ordinary differential equations

A I Asnor\textsuperscript{1,*}, S A M Yatim\textsuperscript{1} and Z B Ibrahim\textsuperscript{2}

\textsuperscript{1}School of Distance Education, Universiti Sains Malaysia, 11800 USM Penang, Malaysia
\textsuperscript{2}Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400, Selangor, Malaysia

*E-mail: asma_asnor@yahoo.com

Abstract. This study is aimed to modify higher order block backward differentiation formulae for dealing with the third order stiff ordinary differential equations (ODEs). Two new solutions are acquired at each step by using a few back values in preceding blocks. Apart from that, the differentiation coefficients will be stored, thus, repetitive calculations of the coefficients can be avoided as well as computational cost can also be optimized. A comparative analysis will be presented once the numerical experiments have been conducted to a few considered problems in order to confirm the efficiency of the proposed method with the existing stiff ODE solver. The analysis proves the accuracy and efficiency of the proposed method can be served as another alternative for third order stiff ODEs.

1. Introduction
The solutions to a problem can be obtained from the exact solutions and also approximated solutions. Methods that are used to generate the approximated solutions of the problems are known as numerical methods. The numerical methods are recommended as another option of generating the solutions especially for the more complex problems and the problems that may not have their exact solutions. Until now, scholars have proposed various numerical methods for various kinds of problems to increase the efficiency of the methods while gaining more accurate solutions.

Basically, for solving the problems, the real-world problems in science and engineering are translated into mathematical problems in the form of differential equations (DEs) or a system of DEs. The DE is an equation that consists of a function and one or more derivatives. Since DE is categorized into partial differential equation (PDE) and ordinary differential equation (ODE), therefore, in this paper we concern with the ODE.

Besides, the ODE problems are sometimes facing stiffness phenomenon. The stiff ODE problems are identified when their eigenvalues are large and negative as well as their exact solutions have a term $e^{-\lambda x}$, where $\lambda$ is a positive constant. As the value of $x$ increases, the term $e^{-\lambda x}$ is rapidly decayed to zero.

Therefore, our main focus is specifically on third order stiff ODE problems in which its general form is written as follows and satisfy the stiff conditions as stated earlier,

$$y'' = f(x, y, y', y'') \text{ in the interval } a \text{ to } b. \quad (1)$$
with its initial conditions

\[ y(a) = y_0, \quad y'(a) = y'_0, \quad y''(a) = y''_0. \]

In addition, only certain numerical methods are suitable for solving the stiff problem. Conventional explicit methods are less efficient to handle this problem since they require many steps to produce the solutions. Due to this, computational effort and time required are increased which is very costly for the computation. Thus, implicit methods worked well in dealing with the stiff problems. Based on the prior studies, the block backward differentiation formula (BBDF) method is an implicit method that can solve lower order and higher order stiff ODEs [1]-[3].

Recently, most of the studies of the third order ODEs are focused on dealing with special and non-stiff problems as in [4]-[8]. Since the third order stiff ODEs problem has not been widely studied until now and it is proven that lower order stiff ODEs work effectively with an implicit method such as BBDF, it is, therefore, opens up an idea of proposing modified higher order block backward differentiation formulae (MHO-BBDF(3)) method for dealing with the third order stiff ODEs problem to cope with the issues. We expect that the proposed method will save cost by reducing the computational effort and time.

This paper is arranged in the following manner. Starting from the derivation of the MHO-BBDF(3) in the second section which includes the derivation of the two points and their first and second derivatives, follows the implementation of the MHO-BBDF(3) using Newton’s iteration in the next section. The subsequent section is the significant part of this paper that demonstrates the effectiveness of the proposed method from the analysis made through the experiments conducted. Then, the conclusion of this study is made in the last section.

2. Deriving Modified Higher Order Block Backward Differentiation Formulae

This section will highlight the steps of deriving the proposed method for every step size considered. The step size considered are constant step size, half of the step size and increase the step size to a factor of 1.8. The step-by-step of developing formul\ae\ for the two points are as follows.

Note that \( x_{n+2} - x_{n+1} = h \), \( x_{n+1} - x_n = h \), \( x_n - x_{n-1} = rh \) and \( x_{n+1} - x_{n-2} = rh \) where \( h \) and \( r \) are corresponded to step size and step size ratio respectively. Two new points with their first and second derivatives are yielded at the end of the derivation.

Step 1: Deriving the formul\ae\ of the MHO-BBDF(3) from the following Lagrange polynomial,

\[
P_s(x) = \sum_{j=0}^{4} y(x_{n+2-j}) \prod_{i=0}^{4} \frac{(x - x_{n+2-j})}{(x_{n+2-j} - x_{n+2-i})}
\]  

(2)

Step 2: Replacing \( x \) in polynomial (2) with \( x_{n+1} + s \cdot h \) and subsequently differentiating the polynomial (2) thrice since the order of the problem in equation (1) is 3 to obtain the first, second and third derivatives. All the derivatives are in terms of \( s \), \( h \), and \( r \).

Step 3: Splitting each derivative into 2 equations by substituting the \( s \) term with 0 and 1. All the equations are now in terms of \( h \) and \( r \) only.

Step 4: Substituting the \( r \) term with value 1, 2 and 5/9 in all the derivative’s equations that we have substituted the \( s \) term with 0 in Step 3 will give us the first point of first, second and third derivatives.

Step 5: Repeating the Step 4 by substituting the \( r \) term with value 1, 2 and 5/9 in all the derivative’s equations that we have substituted the \( s \) term with 1 in Step 3. Thus, give us the first second point of first, second and third derivatives.
Consequently, the formulae for first and second points for all the derivatives are written in the following matrix form.

- **Constant step size, \( r = 1 \):**

\[
\begin{bmatrix}
hy'_{n+1} \\
h^2y''_{n+1} \\
h^3y'''_{n+1}
\end{bmatrix}
= 
\begin{bmatrix}
\frac{1}{12} & 0 & 0 \\
-\frac{1}{12} & \frac{1}{2} & 0 \\
\frac{1}{12} & -\frac{3}{4} & \frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
y_{n-2} \\
y_{n-1} \\
y_n
\end{bmatrix}
+ 
\begin{bmatrix}
\frac{5}{6} & 0 & 0 \\
-\frac{5}{6} & \frac{1}{2} & 0 \\
\frac{5}{3} & -\frac{3}{4} & \frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
y_{n+1} \\
y_{n+2} \\
y_{n+3}
\end{bmatrix}
\]

- **Halving the step size, \( r = 2 \):**

\[
\begin{bmatrix}
hy'_{n+1} \\
h^2y''_{n+1} \\
h^3y'''_{n+1}
\end{bmatrix}
= 
\begin{bmatrix}
\frac{1}{80} & 0 & 0 \\
-\frac{1}{80} & \frac{1}{48} & 0 \\
\frac{3}{40} & -\frac{1}{24} & \frac{1}{8}
\end{bmatrix}
\begin{bmatrix}
y_{n-2} \\
y_{n-1} \\
y_n
\end{bmatrix}
+ 
\begin{bmatrix}
\frac{5}{16} & 0 & 0 \\
-\frac{5}{16} & \frac{1}{16} & 0 \\
\frac{8}{3} & -\frac{3}{4} & \frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
y_{n+1} \\
y_{n+2} \\
y_{n+3}
\end{bmatrix}
\]

- **Increasing the step size to a factor of 1.8, \( r = \frac{5}{3} \):**

\[
\begin{bmatrix}
hy'_{n+1} \\
h^2y''_{n+1} \\
h^3y'''_{n+1}
\end{bmatrix}
= 
\begin{bmatrix}
\frac{729}{1900} & 0 & 0 \\
-\frac{729}{1900} & \frac{13851}{6561} & 0 \\
\frac{133}{63423} & -\frac{575}{74358} & \frac{50}{1103}
\end{bmatrix}
\begin{bmatrix}
y_{n-2} \\
y_{n-1} \\
y_n
\end{bmatrix}
+ 
\begin{bmatrix}
\frac{133}{63423} & 0 & 0 \\
-\frac{133}{63423} & \frac{297}{6561} & 0 \\
\frac{25}{1103} & -\frac{25}{1103} & \frac{50}{1103}
\end{bmatrix}
\begin{bmatrix}
y_{n+1} \\
y_{n+2} \\
y_{n+3}
\end{bmatrix}
\]

\]


From the matrix form (3)-(5), the equation of $h^3y_{n+1}^m$ and $h^3y_{n+2}^m$ are rearranged for obtaining the formulae of $y_{n+1}$ and $y_{n+2}$ respectively. Therefore, the third and sixth row of the matrix form (3)-(5) are updated as follows,

- Constant step size, $r = 1$:

$$
\begin{bmatrix}
hy_{n+1}^1 \\
h^2y_{n+1}^2 \\
y_{n+1} \\
y_{n+2} \\
hy_{n+2}^1 \\
h^2y_{n+2}^2
\end{bmatrix}
= 
\begin{bmatrix}
\frac{1}{12} & 1 & -\frac{3}{2} \\
\frac{12}{5} & 3 & 2 \\
\frac{10}{5} & -\frac{3}{2} \\
\frac{4}{5} & 3 & -\frac{3}{2} \\
\frac{11}{5} & 14 & 19 \\
\frac{12}{5} & 3 & 2 \\
\frac{12}{5} & 3 & 2 \\
\frac{3}{5} & 14 & 24 \\
\end{bmatrix}
\begin{bmatrix}
y_{n-2} \\
y_{n-1} \\
y_{n} \\
y_{n+1} \\
y_{n+2} \\
y_{n+1} \\
y_{n+2} \\
\end{bmatrix}
+ 
\begin{bmatrix}
\frac{5}{6} & 1 \\
\frac{11}{3} & 12 \\
\frac{3}{5} & 2 \\
\frac{4}{5} & 12 \\
\frac{26}{3} & 35 \\
\frac{18}{5} & 0 \\
\end{bmatrix}
\begin{bmatrix}
y_{n-1} \\
y_{n} \\
y_{n+1} \\
y_{n+2} \\
y_{n+1} \\
y_{n+2} \\
\end{bmatrix}
+ h^3
\begin{bmatrix}
0 & 0 \\
0 & 0 \\
0 & 0 \\
1 & 0 \\
0 & 0 \\
0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
f_{n+1} \\
f_{n+2} \\
\end{bmatrix}
$$

(6)

- Halving the step size, $r = 2$:

$$
\begin{bmatrix}
hy_{n+1}^1 \\
h^2y_{n+1}^2 \\
y_{n+1} \\
y_{n+2} \\
hy_{n+2}^1 \\
h^2y_{n+2}^2
\end{bmatrix}
= 
\begin{bmatrix}
\frac{1}{80} & 5 & -\frac{15}{16} \\
\frac{120}{3} & 24 & 8 \\
\frac{128}{3} & -\frac{25}{128} \\
\frac{1}{3} & 1 & 3 \\
\frac{30}{7} & 4 & 2 \\
\frac{60}{7} & 6 & 4 \\
\frac{7}{65} & 9 & -\frac{33}{13} \\
\end{bmatrix}
\begin{bmatrix}
y_{n-2} \\
y_{n-1} \\
y_{n} \\
y_{n+1} \\
y_{n+2} \\
y_{n+1} \\
y_{n+2} \\
\end{bmatrix}
+ 
\begin{bmatrix}
\frac{8}{15} & 5 \\
\frac{28}{15} & 23 \\
\frac{15}{128} & 8 \\
\frac{1}{3} & 3 \\
\frac{16}{5} & 23 \\
\frac{15}{3} & 17 \\
\frac{192}{65} & 33 \\
\end{bmatrix}
\begin{bmatrix}
y_{n-1} \\
y_{n} \\
y_{n+1} \\
y_{n+2} \\
y_{n+1} \\
y_{n+2} \\
y_{n+2} \\
\end{bmatrix}
+ h^3
\begin{bmatrix}
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
f_{n+1} \\
f_{n+2} \\
\end{bmatrix}
$$

(7)
- Increasing the step size to a factor of 1.8, \( r = \frac{5}{9} \):

\[
\begin{bmatrix}
hy'_{n+1} \\
h^2y''_{n+1} \\
y'_{n+1} \\
h^2y''_{n+2} \\
y'_{n+2}
\end{bmatrix} = 
\begin{bmatrix}
-729 & 13851 & -133 \\
1900 & -6561 & -50 \\
13300 & 4025 & 50 \\
13300 & -575 & 25 \\
13300 & 4025 & 50 \\
621 & 594 & 644 \\
247 & 65 & 65
\end{bmatrix}^{-1}
\begin{bmatrix}
297 \\
266 \\
133 \\
19 \\
247 \\
0 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
y_{n-2} \\
y_{n-1} \\
y_n \\
y_{n+1} \\
y_{n+2}
\end{bmatrix} + 
\begin{bmatrix}
0 \\
0 \\
19 \\
133 \\
133
\end{bmatrix}
\]

\[
\begin{bmatrix}
y_{n+1} \\
y_{n+2}
\end{bmatrix}
\]

\( (8) \)

3. Implementing Modified Higher Order Block Backward Differentiation Formulae

In this section, Newton’s iteration will be used for the implementation of the MHO-BBDF(3). The general form of formulae in (6), (7) and (8) is given by,

\[
\begin{bmatrix}
hy'_{n+1} \\
h^2y''_{n+1} \\
y'_{n+1} \\
h^2y''_{n+2} \\
y'_{n+2}
\end{bmatrix} = 
\begin{bmatrix}
\beta_1 & \beta_2 & \beta_3 \\
\gamma_1 & \gamma_2 & \gamma_3 \\
\alpha_1 & \alpha_2 & \alpha_3 \\
\tilde{\beta}_1 & \tilde{\beta}_2 & \tilde{\beta}_3 \\
\tilde{\gamma}_1 & \tilde{\gamma}_2 & \tilde{\gamma}_3 \\
\tilde{\alpha}_1 & \tilde{\alpha}_2 & \tilde{\alpha}_3
\end{bmatrix}
\begin{bmatrix}
y'_{n-2} \\
y'_{n-1} \\
y'_{n-2} \\
y'_{n-1} \\
y'_{n-2}
\end{bmatrix} + 
\begin{bmatrix}
\beta_4 & \beta_5 \\
\gamma_4 & \gamma_5 \\
\alpha_4 & \alpha_5 \\
\tilde{\beta}_4 & \tilde{\beta}_5 \\
\tilde{\gamma}_4 & \tilde{\gamma}_5 \\
\tilde{\alpha}_4 & \tilde{\alpha}_5
\end{bmatrix}
\begin{bmatrix}
y_{n-2} \\
y_{n-1} \\
y_{n} \\
y_{n+1} \\
y_{n+2}
\end{bmatrix} + 
f_{n+1}
\]

\( (9) \)

Define \( y_{n+1}^{(i+1)} \) as the \((i+1)\)th iterative value for \( y_{n+1} \) and \( y_{n+2}^{(i)} \) as the \(i\)th iterative value for \( y_{n+2} \). Subtract \( y_{n+1}^{(i+1)} \) from \( y_{n+1}^{(i)} \) and \( y_{n+2}^{(i+1)} \) from \( y_{n+2}^{(i)} \) yield,

\[
\begin{bmatrix}
e^{(i+1)}_{n+1} \\
e^{(i+1)}_{n+2}
\end{bmatrix} = \frac{1}{h}
\begin{bmatrix}
\beta_4 & \beta_5 \\
\beta_4 & \beta_5
\end{bmatrix}
\begin{bmatrix}
e^{(i)}_{n+1} \\
e^{(i)}_{n+2}
\end{bmatrix}
\]

\( (10) \)
\[
\begin{bmatrix}
\dot{e}_{n+1}^\ast \\
\dot{e}_{n+2}^\ast
\end{bmatrix}^{(i+1)} = \frac{1}{\hbar} \begin{bmatrix}
\gamma_4 \\
\gamma_5
\end{bmatrix} \begin{bmatrix}
\dot{e}_{n+1} \\
\dot{e}_{n+2}
\end{bmatrix}^{(i+1)}.
\]

The system of linear equations are expressed in general form as follows,

\[
E = B^{-1}A
\]

where

\[
E = \begin{bmatrix}
e_{n+1} \\
e_{n+2}
\end{bmatrix}^{(i+1)},
\]

\[
B = \begin{bmatrix}
1 - Wh_\dot{\alpha}_1 - Yh^2 \alpha_1 \beta_1 - Z\alpha_1 \gamma_1 & -\theta_1 - Yh^2 \alpha_1 \beta_2 - Z\alpha_1 \gamma_2 \\
-\theta_2 - Yh^2 \alpha_2 \beta_1 - Z\alpha_2 \gamma_1 & 1 - Wh_\dot{\alpha}_2 - Yh^2 \alpha_2 \beta_2 - Z\alpha_2 \gamma_2
\end{bmatrix},
\]

\[
A = \begin{bmatrix}
Y_1 \\
Y_2
\end{bmatrix} + h \begin{bmatrix}
\psi_1 & 0 \\
0 & \psi_2
\end{bmatrix} \begin{bmatrix}
f(y_{n+1}, y'_{n+1}, y''_{n+1}) \\
f(y_{n+2}, y'_{n+2}, y''_{n+2})
\end{bmatrix} + \begin{bmatrix}
B_1 \\
B_2
\end{bmatrix}
\]

\[
W = \frac{\partial f_{n+1,n+2}}{\partial y_{n+1,n+2}},
\]

\[
Y = \frac{\partial f_{n+1,n+2}}{\partial y'_{n+1,n+2}},
\]

\[
Z = \frac{\partial f_{n+1,n+2}}{\partial y''_{n+1,n+2}}.
\]

\[
\begin{bmatrix}
Y_1 \\
Y_2 \\
B_1 \\
B_2
\end{bmatrix} = \begin{bmatrix}
-1 & \theta_1 & y_{n+1} \\
\theta_2 & -1 & y_{n+2} \\
\alpha_1 & \alpha_2 & y_{n-1} + \alpha_3 y_n \\
\alpha_1 & \alpha_2 & y_{n-2} + \alpha_3 y_n
\end{bmatrix}.
\]

Thus, the approximated solutions of \( \begin{bmatrix}
Y_{n+1}^{(i+1)} \\
Y_{n+2}^{(i+1)}
\end{bmatrix}, \begin{bmatrix}
Y_{n+1}^{(i+1)} \\
Y_{n+2}^{(i+1)}
\end{bmatrix}, \text{and} \begin{bmatrix}
Y_{n+1}^{(i+1)} \\
Y_{n+2}^{(i+1)}
\end{bmatrix} \) are calculated as,

\[
\begin{bmatrix}
Y_{n+1}^{(i+1)} \\
Y_{n+2}^{(i+1)}
\end{bmatrix} = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
Y_{n+1}^{(i)} \\
Y_{n+2}^{(i)}
\end{bmatrix} + \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
e_{n+1} \\
e_{n+2}
\end{bmatrix}^{(i+1)}
\]

\[
\begin{bmatrix}
Y_{n+1}^{(i+1)} \\
Y_{n+2}^{(i+1)}
\end{bmatrix} = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
Y_{n+1}^{(i)} \\
Y_{n+2}^{(i)}
\end{bmatrix} + \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
e_{n+1} \\
e_{n+2}
\end{bmatrix}^{(i+1)}
\]

\[
\begin{bmatrix}
Y_{n+1}^{(i+1)} \\
Y_{n+2}^{(i+1)}
\end{bmatrix} = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
Y_{n+1}^{(i)} \\
Y_{n+2}^{(i)}
\end{bmatrix} + \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
e_{n+1} \\
e_{n+2}
\end{bmatrix}^{(i+1)}
\]
4. Numerical experiments

4.1. Test Problems
In this section, three problems of third order stiff ODEs are tested using the direct method, MHO-
BBDF(3) method in the interval $[0, 2]$. Therefore, the problems considered and their equivalent first
order systems are as follows.

**Problem 1 (Artificial Problem)**

$$y'' = -50y'' - 900y' - 5000y, \quad [0, 2],$$
Initial values: $y(0) = -\frac{1}{50}$, $y'(0) = 0$, $y''(0) = -2$,
Exact solution:

$$y = \frac{\cos(10x)e^{-20x}}{50} - \frac{3e^{-10x}}{50} + \frac{\sin(10x)e^{-20x}}{50},$$
Eigenvalues: $-10, -25/20 \pm 10i$.

**Equivalence First Order System**

$$y_1' = y_2, \quad y_2' = y_3, \quad y_3' = -50y_1 - 900y_2 - 5000y_1,$$
Initial values: $y_1(0) = -\frac{1}{50}$, $y_2(0) = 0$, $y_3(0) = -2$,
Exact solution: $y_1 = \frac{25}{3\cos(10x)e^{-20x}} - \frac{3e^{-10x}}{50} + \frac{5\sin(10x)e^{-20x}}{50},$
$y_2 = -\frac{3\cos(10x)e^{-20x}}{50} - \frac{3e^{-10x}}{50} - \frac{5\sin(10x)e^{-20x}}{50},$
$y_3 = 4\cos(10x)e^{-20x} - 6e^{-10x} + 22\sin(10x)e^{-20x}.$

**Problem 2 (Artificial Problem)**

$$y'' = -140y'' - 4900y' - 36000y, \quad [0, 2],$$
Initial values: $y(0) = 0$, $y'(0) = -\frac{4}{50}$, $y''(0) = 1$,
Exact solution:

$$y = \frac{13e^{-40x}}{17e^{-10x}} - \frac{19e^{-90x}}{800},$$
Eigenvalues: $-10, -40, -90$.

**Equivalence First Order System**

$$y_1' = y_2, \quad y_2' = y_3, \quad y_3' = -140y_1 - 4900y_2 - 36000y_1,$$
Initial values: $y_1(0) = 0$, $y_2(0) = -\frac{4}{10}$, $y_3(0) = 1$,
Exact solution: $y_1 = \frac{13e^{-40x}}{500} - \frac{17e^{-10x}}{26e^{-40x}} - \frac{19e^{-90x}}{17e^{-90x}},$
$y_2 = \frac{80e^{-40x}}{208} - \frac{25e^{-10x}}{17e^{-10x}} + \frac{400e^{-90x}}{1539e^{-90x}},$
$y_3 = \frac{5}{8} - \frac{25e^{-10x}}{17e^{-10x}} + \frac{400e^{-90x}}{1539e^{-90x}}.$
Problem 3 [9]

\[ y'' = -12y'' - 36y', \quad \left[0, \frac{2}{5}\right] \]

Initial values: \( y(0) = -\frac{5}{36}, \quad y'(0) = 1, \quad y''(0) = -7, \)

Exact solution: \( y(x) = -\frac{5}{36} e^{-6x} + \frac{1}{6} xe^{-6x}, \)

Eigenvalues: \(-6, -6, 0\)

Equivalence First Order System

\[ y_1' = y_2, \quad y_2' = y_1, \quad y_3' = -12y_3 - 36y_2, \]

Initial values: \( y_1(0) = -\frac{5}{36}, \quad y_2(0) = 1, \quad y_3(0) = -7, \)

Exact solution: \( y_1 = -\frac{5}{36} e^{-6s} + \frac{1}{6} xe^{-6s}, \)

\[ y_2 = e^{-6s} - xe^{-6s}, \quad y_3 = -7e^{-6s} + 6xe^{-6s}. \]

4.2. Results and Discussion

The performance of the direct method that we proposed in this paper is then compared with the results of reduction methods in MATLAB. The results are analysed at tolerance \(10^{-2}, 10^{-3}, 10^{-4}\) and \(10^{-5}\). The following table 1-3 presented the experimental results of the methods for each problem.

By calculating the difference between the approximated solutions of \( y_{n+2} \) from MHO-BBDF(3) and the exact solutions, the errors of the computed solutions are obtained which is given by,

\[
\text{Error}_{\text{max}} = \max_{1 \leq i \leq n} \left| \frac{y_{i} - y_{i}(x)}{A + B(y(x))_{i}} \right|
\]

where \( n \) is the number of equations, \( y_{i} \) is the approximated solutions and \( y(x) \) are the exact solutions.

\( A = 1, \ B = 0, \ A = 1, \ B = 1, \ A = 0, \ B = 1, \) which referring to absolute error test, mixed errors test and relative errors test respectively.

Subsequently, the average errors and maximum errors of the computed solutions are determined as,

\[
\text{Average error} = \frac{\sum_{i=1}^{\text{NTS}} \sum_{i=1}^{n} (\text{Error})_{i}}{n(\text{NTS})},
\]

\[
\text{Maximum error} = \max_{i \leq \text{NTS}} \left( \max_{1 \leq i \leq n} (\text{Error})_{i} \right)
\]

where \( \text{NTS} \) is the number of total steps.

MATLAB’s ODE solvers of ode15s is variable step variable order method based on the numerical differentiation formulas (NDFs) of orders 1 to 5 or backward differentiation formulas (BDFs) and ode23s is Modified Rosenbrock formula of order 2.
### Problem 1

**Table 1.** Table of analysis for Problem 1.

| Tolerance value | MHO-BBDF(3)       | ode15s          | ode23s          |
|-----------------|-------------------|-----------------|-----------------|
| $10^{-2}$       | $4.755827e-5$     | $2.216143e-3$  | $3.508821e-3$  |
| $10^{-3}$       | $2.19936e-5$      | $2.821139e-4$  | $8.387598e-4$  |
| $10^{-4}$       | $9.094237e-6$     | $7.102259e-5$  | $2.174234e-4$  |
| $10^{-5}$       | $3.579864e-6$     | $9.367342e-6$  | $6.397244e-5$  |
| Average error   |                   |                 |                 |
| $10^{-2}$       | $1.714043e-4$     | $8.83673e-2$   | $6.114796e-2$  |
| $10^{-3}$       | $9.714458e-5$     | $6.123035e-3$  | $1.474177e-2$  |
| $10^{-4}$       | $3.908638e-5$     | $1.873861e-3$  | $3.368918e-3$  |
| $10^{-5}$       | $1.488362e-5$     | $2.276728e-4$  | $7.448675e-4$  |
| Maximum error   |                   |                 |                 |
| $10^{-2}$       | $0.000906$        | $0.017430$      | $0.012471$      |
| $10^{-3}$       | $0.001356$        | $0.022320$      | $0.019886$      |
| $10^{-4}$       | $0.001851$        | $0.023188$      | $0.036235$      |
| $10^{-5}$       | $0.002841$        | $0.031086$      | $0.066972$      |
| Number of total steps |     |                 |                 |
| $10^{-2}$       | $28$              | $71$            | $42$            |
| $10^{-3}$       | $41$              | $100$           | $87$            |
| $10^{-4}$       | $68$              | $132$           | $156$           |
| $10^{-5}$       | $116$             | $150$           | $266$           |
| Time (s)        |                   |                 |                 |
| $10^{-2}$       | $0.000880$        | $0.013960$      | $0.012877$      |
| $10^{-3}$       | $0.001119$        | $0.021049$      | $0.022417$      |
| $10^{-4}$       | $0.001660$        | $0.031990$      | $0.033817$      |
| $10^{-5}$       | $0.002184$        | $0.036133$      | $0.050298$      |

### Problem 2

**Table 2.** Table of analysis for Problem 2.

| Tolerance value | MHO-BBDF(3)       | ode15s          | ode23s          |
|-----------------|-------------------|-----------------|-----------------|
| $10^{-2}$       | $4.198994e-5$     | $3.456765e-4$  | $8.124697e-4$  |
| $10^{-3}$       | $2.075789e-5$     | $6.850130e-5$  | $2.011132e-4$  |
| $10^{-4}$       | $8.943979e-6$     | $6.879320e-6$  | $5.354273e-5$  |
| $10^{-5}$       | $3.499914e-6$     | $3.735521e-6$  | $1.911201e-5$  |
| Average error   |                   |                 |                 |
| $10^{-2}$       | $2.073534e-4$     | $5.804675e-3$  | $9.050251e-3$  |
| $10^{-3}$       | $9.555552e-5$     | $7.774334e-4$  | $2.123363e-3$  |
| $10^{-4}$       | $3.954946e-5$     | $8.368060e-5$  | $4.909736e-4$  |
| $10^{-5}$       | $1.501169e-5$     | $4.667174e-5$  | $1.449325e-4$  |
| Maximum error   |                   |                 |                 |
| $10^{-2}$       | $24$              | $70$            | $42$            |
| $10^{-3}$       | $41$              | $87$            | $80$            |
| $10^{-4}$       | $69$              | $110$           | $143$           |
| $10^{-5}$       | $116$             | $132$           | $233$           |
| Number of total steps |     |                 |                 |
| $10^{-2}$       | $0.000880$        | $0.013960$      | $0.012877$      |
| $10^{-3}$       | $0.001119$        | $0.021049$      | $0.022417$      |
| $10^{-4}$       | $0.001660$        | $0.031990$      | $0.033817$      |
| $10^{-5}$       | $0.002184$        | $0.036133$      | $0.050298$      |
| Time (s)        |                   |                 |                 |
| $10^{-2}$       | $0.000880$        | $0.013960$      | $0.012877$      |
| $10^{-3}$       | $0.001119$        | $0.021049$      | $0.022417$      |
| $10^{-4}$       | $0.001660$        | $0.031990$      | $0.033817$      |
| $10^{-5}$       | $0.002184$        | $0.036133$      | $0.050298$      |
Table 3. Table of analysis for Problem 3.

| Tolerance value | MHO-BBDF(3) | ode15s | ode23s |
|------------------|-------------|-------|--------|
| 10^{-2}          | 7.553616e-4| 1.834832e-2| 1.757528e-3|
| 10^{-3}          | 2.836838e-4| 3.770650e-4| 4.749910e-4|
| 10^{-4}          | 9.590780e-5| 4.595432e-5| 1.304148e-4|
| 10^{-5}          | 3.233593e-5| 6.546990e-6| 3.749517e-5|

| Average error    |             |       |        |
|------------------|-------------|-------|--------|
| 10^{-2}          | 1.960749e-3| 5.507667e-2| 2.687281e-2|
| 10^{-3}          | 7.592491e-4| 7.134088e-3| 5.586032e-3|
| 10^{-4}          | 2.732023e-4| 1.087791e-3| 1.164319e-3|
| 10^{-5}          | 9.264826e-5| 1.272246e-4| 2.541921e-4|

| Maximum error    |             |       |        |
|------------------|-------------|-------|--------|
| 10^{-2}          | 27          | 50    | 30     |
| 10^{-3}          | 47          | 56    | 52     |
| 10^{-4}          | 79          | 64    | 89     |
| 10^{-5}          | 138         | 82    | 142    |

| Number of total steps |       |       |        |
|-----------------------|-------|-------|--------|
| 10^{-2}               | 0.000856 | 0.060800  | 0.036726 |
| 10^{-3}               | 0.001393 | 0.070820  | 0.039448 |
| 10^{-4}               | 0.001681 | 0.075188  | 0.065671 |
| 10^{-5}               | 0.002020 | 0.213785  | 0.103270 |

Table 1-3 presented the numerical results for Problem 1-3. By conducting the numerical experiment with various tested problems, the efficiency of the proposed method, MHO-BBDF(3) can be verified. The data shows that the MHO-BBDF(3) method manage to reduce the number of total steps taken for the computation for most of the problems as compared to the reduction method, ode15s and ode23s. The reduction methods needed more total steps at every tolerance value which took longer computational time. However, as we applied the proposed method for solving the considered problems, the computational time and effort are reducing as well. Furthermore, when analysing all the errors, the errors produced by MHO-BBDF(3) are the least and within the tolerance value. This means that the errors are acceptable. Therefore, the approximated solutions generated by MHO-BBDF(3) are more accurate since the solutions are converged to the exact solutions.

5. Conclusion
The concern on the third order stiff ODEs opens up the idea of developing a new method that manages to apply the direct integration approach. The method that has been developed in this paper is a modified higher order block backward differentiation formula. After being analysed, it is found that the MHO-BBDF(3) method solved the problems effectively since the computational effort and time can be minimized as well as the accuracy of the approximated solutions has also improved. As a whole, the proposed method can be recommended as another option for finding the direct solutions of the third order stiff ODEs.

Acknowledgments
The authors would like to thank Universiti Sains Malaysia for supporting this research under short term grant USM, 304/PJJAUH/6315189.

References
[1] Zainuddin N and Ibrahim Z B 2017 24th National Symposium on Mathematical Sciences vol 1870 p 050009
[2] Nasarudin A A, Ibrahim Z B and Rosali H 2020 Symmetry 12 952
[3] Asnor A I, Yatim S A M and Ibrahim Z B 2019 *ASM Sc. J* **12** 33-40
[4] Adeyeye O and Omar Z 2019 *IAENG International Journal of Applied Mathematics* **49**
[5] Allogmany R and Ismail F 2020 *Mathematics* **8** 1771
[6] Rajabi M, Ismail F and Senu N 2016 *Proc. of the 2nd International Conference on Mathematical Sciences and Statistics* vol 1739 p 020036
[7] Lee K C, Senu N, Ahmadian A, Ibrahim S N I and Baleanu D 2020 *Alexandria Engineering Journal* **59** 2449-67
[8] Turaci M Ö and Özdemir M 2021 *Numerical Methods for Partial Differential Equations* **37** 987-997
[9] Zill D 2009 *A First Course in Differential Equations with Modelling Applications* (Belmont, CA: Brookes-Cole)