Variable Step Three-Point Block Methods for Solving Stiff Ordinary Differential Equations

MAHFUZAH MAHAYADIN\textsuperscript{a}, KHAIRIL ISKANDAR OTHMAN\textsuperscript{b} & ZARINA BIBI IBRAHIM\textsuperscript{c}

\textsuperscript{a,b} Department of Mathematics, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, MALAYSIA

\textsuperscript{c} Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, MALAYSIA

mahfuzah347@gmail.com, khairil@tmsk.uitm.edu.my, zarinabb@science.upm.edu.my

Abstract: - In this paper, we consider the numerical solution of first order stiff ordinary differential equations (ODEs) by a class of variable step size three points Block Backward Differentiation Formulas (BBDFs) methods. The basic formulation using three back values and the step size controlling strategy is described. In order to optimize the performance in terms of precision and the computational time, all the coefficients of the method will be stored in simplified form to avoid the repetitive computation of the coefficients as the step changes in the integration interval. Numerical results show the 3-point BBDF method gives better accuracy compared to the existing ODE solver, ode15s when solving ODEs problems.

Key-Words: - block; stiff, ordinary differential equations; backward differentiation formulas

1 Introduction

Ordinary differential equations (ODEs) often exist in various applied fields of science and engineering. It is widely used to model problems of elasticity theory, hydrodynamics, fluid dynamics, radioactive decay problem and geophysics.

We consider the following systems of $s$ first order ODEs,

\[
y' = f(x, y(x)), \quad y(a) = \eta, \quad (1)
\]

where \( y(x) = [y_1(x), y_2(x), \ldots, y_s(x)]^T \),

\( f(x, y) = [f_1(x, y), f_2(x, y), \ldots, f_s(x, y)]^T \), \( a \leq x \leq b \) and \( \eta = [\eta_1, \ldots, \eta_s]^T \) is the vector of initial conditions.

In recent years, much research has been focused on the numerical solution of ODEs problems using block methods. [5] was among the first proposed to block implicit methods and his ideas were extended by [6] using a Runge-Kutta method. [7] and [2] have developed block implicit for one step method.

For solving stiff equations, [3] has derived 2-point and 3-point implicit block method using constant step size and followed by [4] with 2-point implicit block method using variable step size which are based on block BDFs. The derivation then was extended by [1] using 3-point with four back values implicit block method using constant step size for solving stiff problems.

2 Formulation of the Variable Step Size 3-Point BBDF Method

In Fig. 1, three new values of \( y_{n+1}, y_{n+2} \) and \( y_{n+3} \) are computed simultaneously in a block using three back values \( y_{n-2}, y_{n-1} \) and \( y_{n} \). The step size of the computed block is \( 3h \) and the step size of the previous block is \( 2rh \) where \( r \) is the step size ratio. The step size is restricted to constant step size \( (r = 1) \), double \( (r = 0.5) \) and half \( (r = 2) \).

![Fig. 1. 3-point block method of variable step size](image-url)

The interpolating Lagrange polynomial, \( P_n(x) \) at interpolating points \( (x_{n-2}, f_{n-2}), (x_{n-1}, f_{n-1}), \ldots, (x_{n+3}, f_{n+3}) \) is defined as follows,
\[ P_{m}(x) = L_{m,0} f(x_{n+3}) + L_{m,1}(x) f(x_{n+2}) + \ldots + L_{m,m}(x) f(x_{n+3-m}) \]

\[ = \sum_{j=0}^{m} L_{m,j}(x) f(x_{n+3-j}) \]  

(2)

where \( L_{m,j} = \prod_{i \neq j}^{m} \frac{(x-x_{n+2-i})}{(x_{n+3-j} - x_{n+3-i})} \)

for each \( j = 0, 1, \ldots, m \)

Using Lagrange polynomial,

\[ P_{n}(x) = \sum_{j=0}^{n} \frac{y_j}{\prod_{i \neq j}^{n} (x-x_i)} \]

Substituting \( s = \frac{x-x_{n+3}}{h} \) into equation (3) and differentiate once with respect to \( s \) gives,

\[ P'(x_{n+3} + sh) = \frac{\left[ 135 + 252s^2 + 168s^4 + 12r^3 + 81rs^2 + 174rs + (48s^3 + 35s + 60s^2 + 242s + 22r + 117r) \right] y_{n+3}}{6(3 + r) (3 + 2r)} \]

\[ + \frac{\left[ 54 + 162s^2 + 135s^3 + 12r^4 + 72rs^2 + 126rs + (44s^3 + 15s + 60s^2 + 20r^2 + 12r + 54r) \right] y_{n+2}}{2(2 + r) (2 + 2r)} \]

\[ + \frac{\left[ 27 + 108s^2 + 108s^4 + 12r^5 + 63rs^2 + 90rs + (40s^3 + 35s + 60s^2 + 16r^2 + 6r + 27r) \right] y_{n+1}}{2(1 + r)(1 + 2r)} \]

\[ + \frac{\left[ 18 + 78s^2 + 87s^4 + 12r^6 + 54rs^2 + 66rs + (36s^3 + 35s + 60s^2 + 12r^3 + 4r^2 + 18r) \right] y_{n}}{12 - r^2} \]

\[ + \frac{\left[ 18 + 78s^2 + 87s^4 + 36rs^2 + (44s^3 + 36s^3 + 5s + 12r) \right] y_{n-1}}{r^2 (3 + r) (2 + r)(1 + r)} \]

\[ + \frac{\left[ 18 + 78s^2 + 87s^4 + 36rs^2 + (44s^3 + 36s^3 + 5s + 12r) \right] y_{n-2}}{r^2 (3 + 2r) (2 + 2r)(1 + 2r)} \]  

(4)

By substituting the value \( s = -2, s = -1 \) ands = 0 into equation (4) gives the formula for the first, second and third point respectively. Therefore by letting step size ratio \( r = 1, 2 \) and 0.5 produces the following formulas,

(i) for \( r = 1 \)

\[\begin{align*}
    y_{n+1} &= \frac{1}{10} - \frac{3}{4} \quad 3 - \frac{3}{20} \quad 3 - \\
    y_{n+2} &= -\frac{3}{65} - \frac{4}{13} - \frac{12}{13} - \frac{24}{13} - \frac{12}{65} - \frac{12}{13} - \\
    y_{n+3} &= \frac{12}{137} - \frac{75}{137} - \frac{200}{137} - \frac{300}{137} - \frac{300}{137} - 60 - \\
\end{align*}\]

(ii) for \( r = 2 \)

\[\begin{align*}
    y_{n+1} &= \frac{3}{28} - \frac{5}{4} \quad 75 - \frac{75}{4} - \frac{15}{7} - 30 - \\
    y_{n+2} &= \frac{2}{385} - \frac{3}{55} - \frac{6}{11} - \frac{96}{55} - \frac{96}{385} - \frac{12}{11} - \\
    y_{n+3} &= \frac{15}{1828} - \frac{147}{1828} - \frac{1225}{1828} - \frac{735}{1828} - \frac{3675}{1828} - \\
\end{align*}\]

(iii) for \( r = 0.5 \)

\[\begin{align*}
    y_{n+1} &= \frac{3}{8} - \frac{64}{35} - \frac{3}{3} - \frac{3}{5} - \frac{3}{56} - \frac{3}{2} - \\
\end{align*}\]
The step size strategy is based on the local truncation error, $LTE$ and the error tolerance limit, $TOL$. In the BBDF code, the implementation of step size is limited to halving, constant or doubling. $LTE$ was calculated by taking the difference of the approximate values at the end point of difference order as below

$$LTE = \left\{ (k+1) \right\} y_{n+3} - y_{n+3}$$

where $y_{n+3}$ denote the end point value of order $(k+1)$ and $y_{n+3}$ denote the end point value of order $k$.

The step is considered successful if the local truncation error is less than the tolerance limit ($LTE < TOL$). After a successful step, the step size increment is given by

$$h_{new} = c \times h_{old} \times \left( \frac{TOL}{LTE} \right)^{1/p}$$

and

if $h_{new} > 2 \times h_{old}$ then $h_{new} = 2 \times h_{old}$

where $c$ is the safety factor, $p$ is the order of the method and $h_{old}$ is the step size from the previous block. If the local error criterion is not satisfied ($LTE > TOL$), this step is rejected and the step is repeated with half the current step size.

The errors are defined as the difference between the exact solution, $y(x_{n+3})$, and the approximate solution, $y_{n+3}$.

$$Error_{n+3} = \left| \frac{y(x_{n+3}) - y_{n+3}}{A + B(y_{n+3})} \right|$$

Then, the maximum error is defined as follows,

$$MAXE = \max_{1 \leq i \leq SSTEP} \left\{ \max_{1 \leq j \leq N} Error_{n+3} \right\}$$

where $SSTEP$ is the total number of successful steps and $N$ is the number of equations in the system. $A=1, B=0$ corresponds to the absolute error test, $A=1, B=1$ corresponds to the mixed error test while $A=0, B=1$ corresponds to the relative error test.

### 3 Step Size Control and Error Estimation

### 4 Numerical Results

The validity of the method is illustrated by two (2) examples of ODEs and is compared with the existing ODEs solver, ode15s. We consider the following system of stiff ODEs,

**Problem 1:**

$$y'_1 = 1195y_1 - 1995y_2 \quad y_1(0) = 2$$

$$y'_2 = 1197y_1 - 1997y_2 \quad y_2(0) = -2$$

$$0 \leq x \leq 5$$

Solution:

$$y_1(x) = 10e^{-2x} - 8e^{-800x}$$

$$y_2(x) = 6e^{-2x} - 8e^{-800x}$$

**Problem 2:**

$$y'_1 = -20y_1 - 19y_2 \quad y_1(0) = 2$$

$$y'_2 = -19y_1 - 20y_2 \quad y_2(0) = 0$$

$$0 \leq x \leq 3$$

Solution:

$$y_1(x) = e^{-39x} + e^{-x}$$

$$y_2(x) = e^{-39x} - e^{-x}$$

The numerical results of Problem 1 and Problem 2 are shown in Table 1 and Table 2 respectively.

The notation in the table is defined as follows:

- **TOL**: Tolerance limit
- **MAXE**: Maximum Error
- **BBDF**: 3-point Block Backward Differentiation Formula with three back values
- **ode15s**: A variable order solver based on the numerical differentiation formulas (NDFs)

### Table 1

<table>
<thead>
<tr>
<th>TOL</th>
<th>METHOD</th>
<th>MAXE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-2}$</td>
<td>ode15s</td>
<td>1.52672e-003</td>
</tr>
<tr>
<td></td>
<td>BBDF</td>
<td>5.12461e-004</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>ode15s</td>
<td>3.48626e-003</td>
</tr>
<tr>
<td></td>
<td>BBDF</td>
<td>5.10018e-005</td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td>ode15s</td>
<td>3.55800e-003</td>
</tr>
<tr>
<td></td>
<td>BBDF</td>
<td>8.07048e-006</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>ode15s</td>
<td>5.81929e-003</td>
</tr>
<tr>
<td></td>
<td>BBDF</td>
<td>1.10260e-006</td>
</tr>
</tbody>
</table>
5 Conclusion
The derivation of the variable step size 3-point BBDF has been discussed. The results from the table show that the proposed 3-point BBDF method with three back values give better accuracy compared to existing ODE solver, ode15s.

ACKNOWLEDGEMENT. This research was supported by Universiti Teknologi MARA and under the Fundamental Research Grant Scheme (FRGS), 600-RMI/ST/FRGS 5/3/Fst (11/2011)

References:

Table 2
Numerical Result for Problem 2

<table>
<thead>
<tr>
<th>TOL</th>
<th>METHOD</th>
<th>MAXE</th>
</tr>
</thead>
<tbody>
<tr>
<td>10^-2</td>
<td>ode15s</td>
<td>1.15772e-004</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BBDF 8.66828e-005</td>
</tr>
<tr>
<td>10^-3</td>
<td>ode15s</td>
<td>1.59649e-004</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BBDF 2.42034e-005</td>
</tr>
<tr>
<td>10^-4</td>
<td>ode15s</td>
<td>1.17576e-004</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BBDF 4.51216e-006</td>
</tr>
<tr>
<td>10^-5</td>
<td>ode15s</td>
<td>1.17476e-004</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BBDF 6.63879e-007</td>
</tr>
</tbody>
</table>