CONTINUOUS FOURTH DERIVATIVE BLOCK METHOD FOR SOLVING STIFF SYSTEMS FIRST ORDER ORDINARY DIFFERENTIAL EQUATIONS (ODEs)

by

Ukpebor, L.A and Adoghe, L.O.

Department of Mathematics, Ambrose Alli University, Ekpoma Nigeria.

Emails: adoghelarry@gmail.com, lukeukpebor@gmail.com

Abstract
This paper presents k-point continuous fourth derivative block methods (CFDBM) of order \( k + 4 \) for the solution of stiff systems of ordinary differential equations. The approach uses the collocation and interpolation technique to generate the main continuous fourth derivative method (CFDM) which is then used to obtain the additional methods that are combined as a single block method. Analysis of the methods shows that the method is \( A \)-stable up to order eight. Numerical examples are given to illustrate the accuracy and efficiency of the proposed method.

Keywords: Fourth derivative, Block method, Stiff problems, Continuous,

1.0 Introduction
This study seeks to develop a numerical method for solving stiff initial value problems (IVPs) of first order ordinary differential equations (ODEs) of the form:

\[
y' = f(x, y), y(x_0) = y_0
\]

where \( y \in \mathbb{R}^m, x \in [a, b] \) in ordinary differential equations. We seek a solution of (1.1) in the range \( a \leq x \leq b \) where \( a \) and \( b \) are finite, and we assume that \( f \) satisfies the Lipchitz condition which guarantees that the problem has a unique continuous differentiable solution.

We shall denote this solution by \( y(x) \).

Equation (1.1), occurs in several areas of engineering, sciences and social sciences. Many physical problems are modeled into first order problems. Some of these problems have proved to be either difficult to solve or cannot be solved analytically, hence the need for numerical methods for solving such problems. [1] and [2] posited that there are many methods for solving first order ordinary differential equations. One of the popular methods for solving (1.1) is by Linear Multistep Methods (LMM). This method of solution had been developed in various form such as discrete and continuous linear multistep methods. Continuous linear multistep methods have greater advantages over the discrete methods as they give better error estimation, provide a simplified form of coefficients for further evaluation at different points, and provides solution at all interior points within the interval of integration than the discrete one [3,4]. Second derivative methods have been proposed by [5], [6] and [7]. Recently [7,] and [8] proposed third derivative method of order \( k + 4 \). These methods were implemented in a step-by-step fashion.

In this paper, Continuous Fourth Derivative Block Methods(CFDBM) that will not only be self starting but also possesses good stability properties for effective and efficient numerical integration of problem (1.1) is proposed. The stability and consistency were also examined, so as to confirm the performance of the new method.

2.0 Derivation of the method
We proposed k-step continuous fourth derivative block method form

\[
y(x) = y_{n+k-1} + h \sum_{j=0}^{k} \sigma_j(x)f_{n+j} + h^2 \beta_k(x)p_{n+k} + h^3 \eta_k(x)q_{n+k} + h^4 \delta_k(x)r_{n+k}
\]

(2.1)
for the solution of (1.1) on the interval from \(x_n\) to \(x_{n+k}\), where \(\alpha_j(x), \beta_k(x), \eta_k(x)\) and \(\delta_k(x)\) are the coefficients and \(k\) is the step number and \(h\) is the step length. Interpolation and collocation methods are used in the derivation of the CFDBM. We shall consider, the power series polynomial of the form;
\[
y(x) = \sum_{j=0}^{k} a_j x^j
\]  
(2.2)
as the basis function for approximate solution of (1.1), where \(a_j\)'s are the parameters to be determined. We assume that \(y_{n+j} = y(x_n + jh)\) is the numerical approximation to the analytical solution \(y(x_{n+k}), y'_{n+j} = f(x_{n+j})\) is an approximation to \(y(x_{n+j})\), \(p_{n+k} = f'(x_{n+k}, y(x_{n+k})), q_{n+k} = f''(x_{n+k}, y(x_{n+k})), \) and \(r_{n+k} = f'''(x_{n+k}, y(x_{n+k}))\).

We seek a continuous representation of the CFDBM to approximate the exact solution \(y(x)\) by the interpolating function of the form (2.2)

The first, second, third and fourth derivatives of (2.2) with respect to \(x\) are as given below
\[
f_{n+i} = \sum_{j=0}^{k+4} j a_j x^{j-1}, i = 0(1)k
\]  
(2.3)
\[
p_{n+i} = \sum_{j=0}^{k+4} j(j-1) a_j x^{j-2}, i = k
\]  
(2.4)
\[
a_{n+i} = \sum_{j=0}^{k+4} j(j-1)(j-2) a_j x^{j-3}, i = k
\]  
(2.5)
\[
r_{n+i} = \sum_{j=0}^{k+4} j(j-1)(j-2)(j-3) a_j x^{j-4}, i = k
\]  
(2.6)

Interpolating (2.2) at \(x = x_n\) and Collocating (2.3), (2.4), (2.5) and (2.6) at \(x = x_{n+i}, i = 0(1)k\), results in the system of non-linear equations in the form
\[
\begin{pmatrix}
1 & x_{n+i} & x_{n+i}^2 & x_{n+i}^3 & \cdots & x_{n+i}^{k+4} \\
0 & 1 & 2x_n & 3x_n^2 & 4x_n^3 & \cdots & D_{x_{n+i}}^{k+3} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 1 & 2x_{n+k} & 3x_{n+k}^2 & 4x_{n+k}^3 & \cdots & D_{x_{n+k}}^{k+3} \\
0 & 0 & 2 & 6x_{n+k} & 12x_{n+k}^2 & \cdots & D_{x_{n+k}}^{k+2} \\
0 & 0 & 0 & 6 & 24x_{n+k} & \cdots & D_{x_{n+k}}^{k+1} \\
0 & 0 & 0 & 0 & 24 & \cdots & D_{x_{n+k}}^{k+1}
\end{pmatrix}
\begin{pmatrix}
a_0 \\
a_1 \\
a_2 \\
a_3 \\
a_4 \\
\vdots \\
a_{k+4}
\end{pmatrix}
= \begin{pmatrix}
y_{n+i} \\
f_{n+i} \\
p_{n+k} \\
q_{n+k} \\
r_{n+k}
\end{pmatrix}
\]  
(2.7)

where \(D^k = k + 4, D^k = (k + 4)(k + 3), D^{k+2} = (k + 4)(k + 3)(k + 2), D = (k + 4)(k + 3)(k + 2)(k + 1)\)

Solving equation (2.7) by Gaussian elimination methods yields values of \(a_j\). Substituting the resulting values of \(a_j\) into (2.2) with \(x = x_n + jh\) leads to continuous linear multistep method. The resulting continuous scheme is evaluated at different nodes.

In what follows the block methods for \(k = 3(1)4\) are presented by following the process of derivation above. We have

\[
y_n = y_{n+2} = \frac{263}{945} h f_{n+1} + \frac{731}{420} h f_{n+2} + \frac{209}{105} h f_{n+3} - \frac{7453}{3780} h^2 p_{n+3} - \frac{211}{126} h^2 q_{n+3} + \frac{25}{42} h^3 r_{n+3}
\]
3.0 Stability Analysis of the Methods

3.1 Order and error constant

Following [17, 18], we define the local truncation error associated with the above methods to be the linear difference operator:

\[ L[y(x); h] = \sum_{j=0}^{k} a_j y^{(j)}(x + jh) + h^2 \beta_k y''(x + jh) + h^3 \eta_k y'''(x + jh) + h^4 \delta_k y''''(x + jh) \]  

(3.1)

Assuming that \( y(x) \) is sufficiently differentiable, we can use the term in equation (3.1) as a Taylor series expansion about the point \( x \) to obtain the expression

\[ L[y(x); h] = C_0 y(x) + C_1 h y'(x) + C_2 h^2 y''(x) + C_3 h^3 y'''(x) + \ldots + C_p h^p y^p(x) + \ldots \]

Where the constant coefficients \( C_p, p = 0, 1, 2, 3, \ldots \) are given as follows:

\[
\begin{align*}
C_0 & = \sum_{j=0}^{k} \varphi_j \\
C_1 & = \sum_{j=0}^{k} j \varphi_j - \sum_{j=0}^{k} a_j \\
C_2 & = \frac{1}{2!} \sum_{j=0}^{k} (j^2 \varphi_j - 2a_j) - \beta_k \\
C_3 & = \frac{1}{3!} \sum_{j=0}^{k} (j^3 - 3j^2 a_j) - k \beta_k - \lambda_k \\
C_4 & = \frac{1}{4!} \sum_{j=0}^{k} (j^4 \varphi_j - 4j^3 a_j) - \frac{k^2}{2!} \beta_k - k \lambda_k - \sigma_k
\end{align*}
\]  

(3.2)
According to Henrici (1962), the methods in equation (2.1) has the order $p$ if

$$L[y(x)h] = 0(h^{p-1}), \quad C_0 = C_1 = \cdots = C_p = 0, \quad C_{p+1} \neq 0.$$ 

Therefore, $C_{p+1}$ is the error constant and $C_{p+1}h^{p+1}y^{p+1}(x_n)$ the principal local truncation error at the point $x_n$. It was established from our calculations that the block methods for $k = 3$ and $4$ have orders and error constants as displays in Table (3.1.1) below.

<table>
<thead>
<tr>
<th>$K=3$</th>
<th>$(7, 7, 7)$</th>
<th>(\left(\frac{62y^{(8)}[x]h^8}{33075} - \frac{41y^{(6)}[x]h^6}{156800} - \frac{83y^{(6)}[x]h^6}{4233600}\right)T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K=4$</td>
<td>$(8, 8, 8, 8)$</td>
<td>(\left(-\frac{29y^{(9)}[x]h^9}{12800}, -\frac{19y^{(9)}[x]h^9}{64800}, -\frac{571y^{(9)}[x]h^9}{7257600}, -\frac{59y^{(9)}[x]h^9}{7257600}\right)T)</td>
</tr>
</tbody>
</table>

In what follows, the $k$-step fourth derivative block method can generally be rearranged and rewritten as a matrix finite difference equation of the form:

$$A^{(1)}y_m = A^{(0)}y_{m-1} + A^{(2)}y_{m+1} + h[B^{(0)}f(y_{m-1}) + B^{(1)}f(y_{m})] + h^2[C^{(1)}p(y_m)] + h^3[D^{(1)}q(y_m)] + h^4[E^{(1)}r(y_m)]$$

(3.3)

Where the matrices $r(y_m) = f'''(y_m)q(y_m) = f''(y_m)|p(y_m) = f'(y_m)$ and the matrices $E^{(1)}, D^{(1)}$ and $C^{(1)}$ are strictly diagonal matrix with dimension $k \times k$.

$$A^{(0)} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}, \quad A^{(1)} = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix}, \quad A^{(2)} = \begin{pmatrix} 0 & 1 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \cdots & 0 \end{pmatrix}$$

(3.4)

And the vectors $y_m, y_{m+1}, y_{m+2}, y_{m+3}, p_m, q_m$ and $r_m$ are defined as

$$y_{m+1} = \begin{pmatrix} y_{n+1} \\ y_{n+2} \\ y_{n+3} \\ \vdots \\ y_{n+k} \end{pmatrix}, \quad y_m = \begin{pmatrix} y_{n-k+1} \\ y_{n-k+2} \\ y_{n-k+3} \\ \vdots \\ y_n \end{pmatrix}, \quad f_{m+1} = \begin{pmatrix} f_{n+1} \\ f_{n+2} \\ f_{n+3} \\ \vdots \\ f_{n+k} \end{pmatrix}$$

(3.5)
3.2 Linear Stability

The application of the method (3.3) to the scalar test equations

\[ y' = \lambda y, \ y'' = \lambda^2 y, \ y''' = \lambda^3 y, \ y'''' = \lambda^4 y, \ \text{Re}(\lambda) > 0 \]

yield the stability polynomial

\[ p_{m+1} = \begin{pmatrix} p_{n+1} \\ p_{n+2} \\ p_{n+3} \\ \vdots \\ p_{n+k} \end{pmatrix}, \quad q_{m+1} = \begin{pmatrix} q_{n+1} \\ q_{n+2} \\ q_{n+3} \\ \vdots \\ q_{n+k} \end{pmatrix}, \quad r_{m+1} = \begin{pmatrix} r_{n+1} \\ r_{n+2} \\ r_{n+3} \\ \vdots \\ r_{n+k} \end{pmatrix} \]

Where the \( M(z) \) is the amplification matrix given by

\[ M(z) = -(A_1 - A_2 - B_1 z - C_1 z^2 - D_1 z^3 - E_1 z^4)^{-1}(A_0 + ZB_0) \]

and \( A^{(i)}, B^{(i)}, C^{(i)}, D^{(i)} \) and \( E^{(i)} \) are matrices.

The stability function for the method (3.3) is the polynomial \( \pi(w, z) \) given by

\[ \pi(w, z) = \det[I_k w - M(z)] \]

The region of absolute stability \( RAS \) of SDBM (3.3) is defined by

\[ RAS = \{ z \in \mathbb{C}; |p(M(z))| \leq 1 \} \]

The Boundary locus is used to determine the region of absolutely stability of the continuous fourth derivative block method (CFDBMS).

3.3 Zero Stability

The block method (3.3) is zero stable provided the roots \( R_j, j = 1, \ldots, k \) of the first

Characteristics polynomial \( \rho(R) \) specified by

\[ \rho(r) = \det \left[ \sum_{i=0}^{k} A^{(i)} R^{k-1} \right] = 0, \quad A^0 = I \]

Satisfied \( |R_j| \leq 1, j = 1, \ldots, k \) and for those roots with \( R_j = 1 \) the multiplicity does not exceed 1

3.4 Consistent

The block method (3.3) is consistent if it has order at least one

3.5 Convergent

The block method (3.3) is convergent if and only if it is consistent and zero stable
4.0 Numerical Experiments

To study the efficiency of the block method, some numerical examples to illustrate the accuracy of the methods are presented in this section. Absolute errors of the approximate solution on the partition $\pi_N$ is found using $|y - y(x)|$. Some problems are considered for CFDBM developed in terms of their efficiency and results are compared with those of existing methods. Maple software was used to code the schemes derived and tested on some numerical problems.


**Problem 1:**

$$y'_1 = -y_1 + 95y_2, \quad y_1(0) = 1, y'_2 = -y_1 - 97y_2, \quad y(2) = 1$$

Its exact solutions are given as

$$y_1 = \frac{(95e^{-2x} - 48e^{-95x})}{47}, y_2 = \frac{(48e^{-96x} - e^{-2x})}{47}$$

<table>
<thead>
<tr>
<th>$h$</th>
<th>Methods</th>
<th>$y_1$</th>
<th>$y_2 \times 10^2$ (error)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0625</td>
<td>CH4</td>
<td>0.2735498 (3.0 $\times 10^{-7}$)</td>
<td>-0.2879471 (4.0 $\times 10^{-9}$)</td>
</tr>
<tr>
<td></td>
<td>CH5</td>
<td>0.27554005 (3.0 $\times 10^{-8}$)</td>
<td>-0.2879274 (3.0 $\times 10^{-9}$)</td>
</tr>
<tr>
<td></td>
<td>J-K</td>
<td>0.2735523 (1.0 $\times 10^{-8}$)</td>
<td>-0.2879477 (4.0 $\times 10^{-9}$)</td>
</tr>
<tr>
<td></td>
<td>$F^4$</td>
<td>0.2735503 (3.0 $\times 10^{-7}$)</td>
<td>-0.2879477 (3.1 $\times 10^{-7}$)</td>
</tr>
<tr>
<td></td>
<td>$F^5$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table (4.1) Comparative analysis of result of problem 1 for CFDBM for $k=3$
**Remark:** we observed that the CFDBM of order 7 has been shown to be more efficient and gives a more accurate approximation compared to the method derived in the literature.

**Table (4.1b) Comparative analysis of result of problem 1 for CFDBM for k=4**

<table>
<thead>
<tr>
<th>$h$</th>
<th>Methods</th>
<th>$y_1$</th>
<th>$y_2 \times 10^4$ (error)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0625</td>
<td>CH4</td>
<td>0.2735498(3.0 × 10^{-7})</td>
<td>-0.2879471 (4.0 × 10^{-9})</td>
</tr>
<tr>
<td></td>
<td>CH5</td>
<td>0.27554005(3.0 × 10^{-8})</td>
<td>-0.2879274 (3.0 × 10^{-9})</td>
</tr>
<tr>
<td></td>
<td>J-K</td>
<td>0.2735523 (1.0 × 10^{-8})</td>
<td>-0.2879477 (3.1 × 10^{-9})</td>
</tr>
<tr>
<td></td>
<td>$F^4$</td>
<td>0.27355303 (3.0 × 10^{-7})</td>
<td>-0.2879441 (6.7 × 10^{-9})</td>
</tr>
<tr>
<td></td>
<td>$F^5$</td>
<td>0.2735503 (6.4 × 10^{-9})</td>
<td>-0.2879474 (2.4 × 10^{-8})</td>
</tr>
<tr>
<td></td>
<td>OK6</td>
<td>0.27354657 (3.4 × 10^{-6})</td>
<td>-0.2879464 (4.7 × 10^{-9})</td>
</tr>
<tr>
<td></td>
<td>$F^6$</td>
<td>0.27355005 (1.0 × 10^{-8})</td>
<td>-0.2879469 (4.7 × 10^{-9})</td>
</tr>
<tr>
<td></td>
<td>CFDBM4</td>
<td>0.27354864 (1.3 × 10^{-9})</td>
<td>-0.2879474 (5.0 × 10^{-10})</td>
</tr>
<tr>
<td></td>
<td>F6</td>
<td>0.27355004 (2.0 × 10^{-10})</td>
<td>-0.2879474 (5.0 × 10^{-12})</td>
</tr>
<tr>
<td></td>
<td>Exact solution</td>
<td>0.27355004 (2.0 × 10^{-10})</td>
<td>-0.28794741</td>
</tr>
</tbody>
</table>

| 0.03125| CH4          | 0.27355003 (1.0 × 10^{-8}) | -0.28794742 (1.0 × 10^{-8}) |
|        | J-K          | 0.27355005 (5.0 × 10^{-7}) | -0.28794742 (4.0 × 10^{-7}) |
|        | AB7          | 0.273545505 (4.0 × 10^{-5}) | -0.28794751 (6.0 × 10^{-5}) |
|        | OK6          | 0.27354657 (3.4 × 10^{-6}) | -0.28355004 (3.7 × 10^{-8}) |
|        | $F^4$        | 0.27355005 (1.0 × 10^{-8}) | -0.28794742 (1.0 × 10^{-8}) |
|        | $F^5$        | 0.27355004 (6.3 × 10^{-10}) | -0.28794740 (1.4 × 10^{-10}) |
|        | AF5          | 0.27354958 (4.5 × 10^{-7}) | -0.28794694 (4.7 × 10^{-9}) |
|        | $F^6$        | 0.27355004 (6.0 × 10^{-11}) | -0.2879474 (5.0 × 10^{-10}) |
|        | CFDBM4       | 0.2735500401(2.0 × 10^{-10}) | -0.28794741 |
|        | F6           | 0.27355004 (2.0 × 10^{-10}) | -0.28794741 |
|        | Exact solution| 0.27355004 (2.0 × 10^{-10}) | -0.28794741 |

| 0.05   | OK6          | 0.27354864264 (1.0 × 10^{-6}) | -0.2879459394 (1.0 × 10^{-4}) |
|        | AF5          | 0.27354738 (2.7 × 10^{-6}) | -0.28794461 (1.4 × 10^{-8}) |
|        | $F^6$        | 0.27355004 (3.7 × 10^{-8}) | -0.2879474 (1.4 × 10^{-8}) |
|        | CFDBM4       | 0.2735500400 (5.0 × 10^{-10}) | -0.28794741 (5.0 × 10^{-12}) |
|        | Exact solution| 0.27355004 (2.0 × 10^{-10}) | -0.28794741 |
Remark: we observed that the CFDBM of order 8 has been shown to be more efficient and gives a more accurate approximation compared to the method derived in the literature.

**Problem 2:**

\[ y' + 1001y' + 1000y = 0, y(0) = 1, y'(0) = 1 \]

Exact Solution:  
\[ y(x) = \frac{1001}{999} e^{-x} - \frac{2}{999} e^{-1000x} \]

The problem above is a second order problem, but our method is only capable of handling first order problem, hence there is need to convert it to first order system and before applying our method to solve it. Let \( y = y_1 \) and \( y_1 = y_2 \) the equation becomes

\[ y_2' + 1001y_2' + 1000y_1 = 0, \quad y_1(0) = 1 \]

This can be written in its equivalent system of first order stiff problem as

\[ y_1' = y, \quad y(0) = 1 \]
\[ y_2' = -1001y_1 - 1000y_1, \quad y_1(0) = 1 \]

The exact solutions are given as

\[ y_1(x) = \frac{1001}{999} e^{-x} - \frac{2}{999} e^{-1000x}, \quad y_2(x) = -\frac{1001}{999} e^{-x} + \frac{2000}{999} e^{-1000x} \]

**Table 4.2a: Comparative analysis of result of problem 2 for \( k=3 \)**

<table>
<thead>
<tr>
<th>Step Size h</th>
<th>Method</th>
<th>( y(1) )</th>
<th>Absolute error (y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05</td>
<td>OK6</td>
<td>0.367879436</td>
<td>5.6 x 10^{-8}</td>
</tr>
<tr>
<td></td>
<td>F5</td>
<td>0.367879440</td>
<td>5.2 x 10^{-9}</td>
</tr>
<tr>
<td></td>
<td>AG6</td>
<td>0.36787846</td>
<td>1.4 x 10^{-8}</td>
</tr>
<tr>
<td></td>
<td>AF5</td>
<td>0.36787930</td>
<td>1.8 x 10^{-7}</td>
</tr>
<tr>
<td></td>
<td>*F6</td>
<td>0.36787840</td>
<td>4.4 x 10^{-9}</td>
</tr>
<tr>
<td></td>
<td>CFDBM3</td>
<td>0.368615936537513</td>
<td>1.650 x 10^{-11}</td>
</tr>
<tr>
<td>0.125</td>
<td>F6</td>
<td>0.367879442</td>
<td>2.7 x 10^{-8}</td>
</tr>
<tr>
<td></td>
<td>CFDBM3</td>
<td>0.368615936075075</td>
<td>4.74088 x 10^{-10}</td>
</tr>
</tbody>
</table>

**Remark:** The numerical results in table (4.2a) show that CFDBM compares favourably with method in the literature.

**Table 4.2b: Comparative analysis of result of problem 2 for \( k=4 \)**

<table>
<thead>
<tr>
<th>Step Size h</th>
<th>Method</th>
<th>( y(1) )</th>
<th>Absolute error (y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05</td>
<td>OK6</td>
<td>0.367879436</td>
<td>5.6 x 10^{-8}</td>
</tr>
<tr>
<td></td>
<td>F5</td>
<td>0.367879440</td>
<td>5.2 x 10^{-9}</td>
</tr>
<tr>
<td></td>
<td>AG6</td>
<td>0.36787846</td>
<td>1.4 x 10^{-8}</td>
</tr>
<tr>
<td></td>
<td>AF5</td>
<td>0.36787930</td>
<td>1.8 x 10^{-7}</td>
</tr>
<tr>
<td></td>
<td>*F6</td>
<td>0.36787840</td>
<td>4.4 x 10^{-9}</td>
</tr>
<tr>
<td></td>
<td>CFDBM4</td>
<td>0.368623137744890</td>
<td>7.20120 x 10^{-6}</td>
</tr>
<tr>
<td>0.125</td>
<td>F6</td>
<td>0.367879442</td>
<td>2.7 x 10^{-8}</td>
</tr>
<tr>
<td></td>
<td>CFDBM4</td>
<td>0.368570858598249</td>
<td>4.50780 x 10^{-9}</td>
</tr>
</tbody>
</table>

**Remark:** The numerical results in table (4.2b) show that CFDBM compares favourably with method in the literature.

**5.0 Conclusion**

288
A newly derived family of Continuous Fourth Derivative Block Method has been developed for the solution of stiff systems of ordinary differential equations and used to simultaneously solve (1.1) directly without the need for starting values or predictors. The efficiency of the CFDBM has been demonstrated on some standard numerical examples. Details of the numerical results are displayed in Table (4.1) and (4.2).

6.0 References
Onumanyi P, Awoyemi DO, Jator SN, Sirisena UW. New linear multistep methods with continuous coefficients for first order initial value