The Mathematica Kernel Programming Codes Designed for Implementing Block Milne's Device

J. G. Oghonyon, *Member, IAENG*, N. A. Omoregbe, S. A. Bishop

Abstract- **In this article, we propose the Mathematica kernel programming codes designed for implementing block Milne's device employing an expounded trigonometrically fitted method. Block Milne's device is an extraction of Adam's family constructed via expounded trigonometrically fitted method. We execute the Mathematica kernel programming codes of block Milne's device in a block by block mode. This proficiencies of scientific computing have great advantages of easy computation, speed, faster convergence and accuracy. Other numerical gains of block Milne's device includes; changing the step-size, deciding the convergence criteria and control errors. Additionally, the Mathematica kernel programming codes for implementing block Milne's device is performed on some special problems to demonstrate the accuracy and efficiency.**

Index Terms- **Mathematica kernel, expounded trigonometrically fitted method, block Milne's device, convergence criteria, principal local truncation error.**

I. INTRODUCTION

Mathematica is the invention of Stephen Wolfram, a theoretic scientist who has committedly established essential impacts to maths and computing. Wolfram identifies Mathematica as "the world's only fully integrated environment for technical computing". See [13].

Mathematica is a computing device (information processing system) that is used to execute numerical, symbolical and graphic computing. As described by the creators, Wolfram Research, Inc. Mathematica is "a system for doing mathematics by computer". Mathematica is distinct from previous computer programming language that are utilized by economic expert (FORTRAN, BASIC, PASCAL, C. etc.). It is a translated computing language, i.e., to each one input signal command develops quick output signal. Altho Mathematica can be applied as a computing programming language, its high-altitude construction is more befitting for executing advanced math operations via the use of inherent mathematical functions. For instance, Mathematica can determine limits, differentiations, integrations, determinants, plotting of graphs and carries out symbolical computings as cited in [15].

Manuscript submitted June 22, 2017; revised July 20, 2017. This work was supported by Covenant University Centre for Research, Innovation and Discovery (CUCRID, Ota, Ogun State, Nigeria.

J. G. Oghonyon, Department of Mathematics, College of Science and Technology, Covenant University, P.M.B. 1023, Ota, Ogun State, Nigeria (corresponding author phone no.: +234-8139724200; godwin.oghonyon@covenantuniversity.edu.ng).

N. A. Omoregbe, Department of Mathematics, College of Science and Technology, Covenant University, P.M.B. 1023, Ota, Ogun State, Nigeria (nicholas.omoregbe@covenantuniversity.edu.ng).

S. A. Bishop Sheila Amina, Department of Mathematics, College of Science and Technology, Covenant University, P.M.B. 1023, Ota, Ogun State, Nigeria (Sheila.bishop@covenantuniversity.edu.ng).

Mathematica is divided into two parts: Mathematica front end (notebook) and the Mathematica kernel (kernel). The front end admits input signal, demonstrates output signal. See [15] for more info. The front end is the most important that allows the user to interact with the system for the aim of carrying out calculus and to preserve them reprocess or for reference point. The Mathematica kernel is the unseeable part of the computing program that carries out all the computations as discussed in [31].

II. STATEMENT OF THE PROBLEM

This study considers special problems with exceptional property whose approximate solution is already known in ahead of time. Such peculiar problems can be of the form

$$y'' = f(x,y), \quad y(x_0) = y_0, \quad y'(x_0) = y_0' \text{ for } x \epsilon [x_0,\ X], \tag{1}$$

where $f: R \times R^d \to R^d$, d is the dimension of the physical system as seen in [8], [12], [18].

Par (1) which is known to satisfy both [5], [17] originates from fields of scientific discipline and applied science such as Newtonian mechanics, uranology, quantum theory, control theory, electric circuit and biological science. Diverse scientific computing techniques founded on trigonometrically fitted method whose result are recognized beforehand to represent periodical/oscillating occurrence having a recognized frequence and belonging to a family of technics established on trigonometric multinomial formulated by [12], [18] is in particular set aside. Several authors have suggested and implemented par (1) to generate the desired result. Among them are [23]-[25] executed all computings utilizing a composed computer code in Matlab. Again, [9], [26]-[27] carries out numeric experimentation applying a written cipher in Mathematica 10. 0 to demonstrate the accuracy and effectiveness of the technics, while [7] executed all mathematical computings on a PC computing device initiated by running PYTHON.

The main goal of this research work is focus on developing the Mathematica kernel programming codes designed for implementing block Milne's device to compute (1). Other benefits of scientific computing and block Milne's device have been enlisted in the abstract. Block Milne's device is considered as an extensive view of the block predictor-corrector pair on account of the more outstanding numerical vantages as stated earlier. This include some remarkable components such as; Adams type family, block predictor-corrector pair of the same order and principal local truncation errors as sited [4], [10]-[11], [18]-[19], [29]-[30].

Definition 1- A trigonometric multinomial (polynomial) with the highest degree k can be defined as

$$y(x) = a_0 + \sum_{j=1}^{k}\big(a_j \sin(jwx) + a_{j+1}\cos(jwx)\big), \qquad (2)$$

where $a_0, a_j \; and \; a_{j+1}$ are the unknown coefficients and $y(x) \; is \; C(2\pi)$ whenever it is continuous with period 2π. See [6], [32] for details.

Definition 2- $b-block, r-point\ method$. If k refers to the block size and h is the pace size, then block size in time is rh**.** Let $m = 0,1,2,\ldots$ form the block number and let $n = mr$, then the $b-block$, $r-point$ method can be composed in the next general class:

$$Y_\tau = \sum_{v=1}^{b} A_v Y_{\tau-v} + h\sum_{v=0}^{b} B_v F_{\tau-v}$$
(3)

where

$Y_\tau = [y_{n+1}, \ldots, y_{n+i}, \ldots, y_{n+r}]^T$,
$F_\tau = [f_{n+1}, \ldots, f_{n+i}, \ldots, f_{n+r}]^T$
A_v and B_v are $r \times r$ constants matrices. See [14], [30].

Hence, starting out from the supra explanation, a block method has the computing benefits that for each practical application program, the terminate product is assessed to a greater extent at the same time. The numerate of points banks on the construction of the block method. Thusly, utilizing these technics can allow for quicker and faster outcomes of the problem which can be handled to generate the sought after accuracy. Check out [20]-[22], [29]-[30] for more info.

Theorem (Weierstrass Approximation Theorem)

Let $f: R \to R$ be continuous and 2π −periodic. Then for each $\varepsilon > 0$, there exists a trigonometric polynomial $P(x) = \sum_{j=-n}^{k} c_j\, e^{ijx}$ such that for all $x, |f(x) - P(x)| < \varepsilon$. Tantamountly, as for any such f, there must exist a successive polynomials such that $P_n \to f$ in a uniform manner on R as cited in [5].

Theorem (Existence and Uniqueness)

Let $f(x,y)$ be defined and continuous for all points (x,y) in the region D defined by $a \le x \le b$, $-\infty < y < \infty$, where a and b are finite, and let there exists a constant L such that for any $x \in [a,b]$ and any two numbers y and $\bar{y}$,

$$|f(x,y) - f(x,\bar{y})| \le L|y - \bar{y}|.$$

This precondition is recognized as *Lipschitz* condition. Then there is precisely a single function $y(x)$ having the succeeding quadruplet attributes.

- $y(x)$ is continuous and differentiable for $x \in [a,b]$,
- $y''(x) = f(x, y(x)), x \in [a,b]$,
- $y(x_0) = y_0$,
- $y'(x_0) = y_0'$. See [17].

The rest of this research work is considered in details as adopts: in Section 2 The Mathematica Kernel Programming Codes of the Materials and Methods. Section 3 The Mathematica Kernel Programming Codes of the Numerical Results and Discussion. Section 4 Conclusion as mentioned in [2], [29]-[30].

III. THE MATHEMATICA KERNEL PROGRAMMING CODES OF THE MATERIALS AND METHODS

This section considers the formulation of the Mathematica kernel programming codes of block Milne's device. In this section, the goal to be attained is to formulate block Milne's device. Block Milne's device is a merger of Adams-Bashforth $q-step$ (predictor) method and Adams-Moulton $q-1-step$ (corrector) method of the like order. This merger can be of the class

$$y(x) = \sum_{j=0}^{k} \alpha_i\, y_{n-i} + h^2 \sum_{j=0}^{k} \beta_i(u) f_{n-i}, \qquad (4)$$

$$y(x) = \sum_{j=0}^{k} \alpha_i\, y_{n-i} + h^2 \sum_{j=1}^{k} \boldsymbol{\beta}_i^*(u) f_{n+i}. \qquad (5)$$

Pars (4) and (5) constitutes the Adams class of Block Milne's device with$u = wh$, $\beta_j(u)$, $j = 0, 1, 2$ comprising features that bank on the changing step-size and frequence. Mentioning that y_{n+j} is the numerical approximate to the analytical results $y(x_{n+j})$ i.e. $y(x_{n+j}) \approx y_{n+j}$, and $f(x_{n+j}, y_{n+j}) \approx f_{n+j}$ possessing $j = 0, 1, 2$. To arrive at pars (4) and (5), the trigonometrically fitted method is rewritten as the expounded trigonometrically fitted method in which operates by expecting to approximate the analytical result $y(x)$ on clear cut time intervals of $[x_n, x_{n-j}]$ via the interpolating subprogram of the form (2)

$$y(x) = \sum_{j=0}^{k} a_j + \sum_{j=1}^{k}\big(a_j \sin(jwx) + a_{j+1}\cos(jwx)\big).$$
(6)

Retooling (6) brings about the expounded trigonometrically fitted method which can represented in Mathematica kernel programming codes as

$$y[x\_] = a[0] + a[1]\frac{(x - x[n])}{h} + a[2] + a[3]\frac{(x - x[n])^3}{h^3} + a[4]\left(\frac{w(x-x[n])}{h} - \frac{w^3(x-x[n])^3}{6h^3} + \frac{w^5(x-x[n])^5}{120h^5}\right) + a[5]\left(1 - \frac{w^2(x-x[n])^2}{2h^2} + \frac{w^4(x-x[n])^4}{24h^4}\right), \qquad (7)$$

where a_0, a_1, a_2, a_3, a_4 and a_5 are invariants which is needed to be determine in a peculiar way. Presume the precondition that method (6) agrees with the analytical result at the time interval x_n, x_{n-j} to become the approximate of

$$y(x_n) \approx y_n, \quad y(x_{n-k}) \approx y_{n-k}. \qquad (8)$$

Requiring that the estimating function (7) gratifies problem (1) at the points $x_{i+k}, k = 0, 1, 2$ to develop the succeeding approximates as

$$y''(x_{n+k}) \approx f_{n+k}], \qquad k = 0, 1, 2. \qquad (9)$$

Combining the approximates of (8) and (9) will give rise to a fivefold systems of equation which results to Ax=b. Computing the systems of equation will yield the block Milne's device (block predictor-corrector pair) represented as the Mathematica kernel programming codes

$$matrixa = \left\{\begin{array}{c} \{1,0,0,0,01\}, \\ \left\{1,-1,1,-1,-w+\frac{w^3}{6}-\frac{w^5}{120},1-\frac{w^2}{2}+\frac{w^4}{24}\right\}, \\ \{0,0,2,0,0,-w^2\}, \\ \left\{0,0,2,-6,w^3-\frac{(w^5)}{6},-w^2+\frac{w^4}{2}\right\}, \\ ,\left\{0,0,2,-12,(2w^3)-\frac{(4w^5)}{3},-w^2+(2w^4)\right\}, \\ \left\{0,0,2,-18,(3w^3)-\frac{(9w^5)}{2},-w^2+\frac{(9w^4)}{2}\right\} \end{array}\right\};$$

$$b=\{y[n],y[n-1],f[n],f[n-1],f[n-2],f[n-3]\};$$

$$\{c,d,l,q,t,u\}=Inverse[matrixa].b \qquad (10)$$

$$matrixa = \left\{\begin{array}{c} \{1,0,0,0,01\}, \\ \left\{1,-1,1,-1,-w+\frac{w^3}{6}-\frac{w^5}{120},1-\frac{w^2}{2}+\frac{w^4}{24}\right\}, \\ \left\{0,0,2,-12,(2w^3)-\frac{(4w^5)}{3},-w^2+(2w^4)\right\}, \\ \left\{0,0,2,6,(-w^3)+\frac{(w^5)}{6},-w^2+\frac{(w^4)}{2}\right\}, \\ \left\{0,0,2,12,(-2w^3)+\frac{(4w^5)}{3},-w^2+(2w^4)\right\}, \\ \left\{0,0,2,18,(-3w^3)+\frac{(9w^5)}{2},-w^2+\frac{(9w^4)}{2}\right\} \end{array}\right\};$$

$$b=\{y[n],y[n-1],f[n-2],f[n+1],f[n+2],f[n+3]\};$$

$$\{c,d,l,q,t,u\}=Inverse[matrixa].b, \qquad (11)$$

to obtain $a_k, k=0,1,2,3,4,5$ and replacing the values of $a_k's$ into (6) will obtain the continuous block Milne's device as

$$y[x\_]=\left(1+\frac{(x-x[n])}{h}\right)y[n]+\left(-\frac{(x-x[n])}{h}\right)y[n-1]+$$

$$\left(-\frac{2}{w^4}-\frac{\left((-12w^5)+\frac{(97w^9)}{30}\right)}{(12w^9)}+\frac{(12w^7+6w^9)}{(12w^9)}\frac{(x-x[n])^2}{h^2}+\frac{\left((2w^7)+\frac{(11w^9)}{3}\right)}{(12w^9)}\frac{(x-x[n])^3}{h^3}+\left(\frac{1}{w^5}\right)\frac{(x-x[n])^4}{h^4}+\left(\frac{2}{w^4}\right)\frac{(x-x[n])^5}{h^5}\right)f[n]h^2+$$

$$\left(\left(\frac{5}{w^4}\right)+\frac{\left((36w^5)+\frac{(19w^9)}{5}\right)}{(12w^9)}\frac{(x-x[n])}{h}-\left(\frac{5}{(2w^2)}\right)\frac{(x-x[n])^2}{h^2}+\frac{((-6w^7)-(6w^9))}{(12w^9)}\frac{(x-x[n])^3}{h^3}--\left(\frac{3}{w^5}\right)\frac{(x-x[n])^4}{h^4}-\left(\frac{5}{w^4}\right)\frac{(x-x[n])^5}{h^5}\right)f[n-1]h^2+$$

$$\left(\left(\frac{-4}{w^4}\right)+\frac{\left((-36w^5)-\frac{(13w^9)}{10}\right)}{(12w^9)}\frac{(x-x[n])}{h}+\left(\frac{2}{w^2}\right)\frac{(x-x[n])^2}{h^2}+\frac{((6w^7)+(3w^9))}{(12w^9)}\frac{(x-x[n])^3}{h^3}+\left(\frac{3}{w^5}\right)\frac{(x-x[n])^4}{h^4}+\left(\frac{4}{w^4}\right)\frac{(x-x[n])^5}{h^5}\right)f[n-2]h^2+$$

$$\left(\left(\frac{1}{w^4}\right)+\frac{\left((12w^5)+\frac{(4w^9)}{15}\right)}{(12w^9)}\frac{(x-x[n])}{h}-\left(\frac{1}{2w^2}\right)\frac{(x-x[n])^2}{h^2}+\frac{\left((-2w^7)-\frac{(2w^9)}{3}\right)}{(12w^9)}\frac{(x-x[n])^3}{h^3}-\left(\frac{1}{w^5}\right)\frac{(x-x[n])^4}{h^4}-\left(\frac{1}{w^4}\right)\frac{(x-x[n])^5}{h^5}\right)f[n-3]h^2, \qquad (12)$$

$$y[x\_]=\left(1+\frac{(x-x[n])}{h}\right)y[n]+\left(-\frac{(x-x[n])}{h}\right)y[n-1]+$$

$$\left(-\frac{1}{(5w^4)}+\frac{\left(12w^5+\frac{(323w^9)}{30}\right)}{(120w^9)}+\frac{(12w^7+6w^9)}{(120w^9)}\frac{(x-x[n])^2}{h^2}+\right.$$

−2w7−11w93120w9x−xn3h3−110w5x−xn4h4+15w4x−xn5h5

$$f[n-2]h^2+\left(\left(\frac{1}{w^4}\right)+\frac{\left(-120w^5+\frac{(382w^9)}{3}\right)}{(120w^9)}\frac{(x-x[n])}{h}+\right.$$

−60w7+120w9120w9x−xn2h2−20w7−40w93x−xn3h3+1w5x−xn4h4−1w4x−xn5h5

$$f[n+1]h^2+\left(\left(\frac{-1}{w^4}\right)+\frac{\left(180w^5-\frac{(217w^9)}{2}\right)}{(120w^9)}\frac{(x-x[n])}{h}+\right.$$

60w7−90w9120w9x−xn2h2+−30w7+25w9120x−xn3h3−32w5x−xn4h4+1w4x−xn5h5

$$f[n+2]h^2+\left(\left(\frac{1}{5w^4}\right)+\frac{\left(-72w^5+\frac{152w^9}{5}\right)}{(120w^9)}\frac{(x-x[n])}{h}+\right.$$

−12w7+24w9120w9x−xn2h2+12w7−8w9120x−xn3h3−35w5x−xn4h4−15w4x−xn5h5 fn+3h2

(13)

Assessing the continuous block Milne's device of par (12) and (13) at some preferred points of x_{n+k}, $k=1,2,3$ will formulate the block Milne's device as

$$y[x\_]=y[n]+y[n-1]+h^2(\beta_0(w,x)f[n]+\beta_1(w,x)f[n-1]+$$

β2w,xfn−2+β3w,xf[n−3],

(14)

$$y[x\_]=y[n]+y[n-1]+h^2(\beta_0(w,x)f[n-2]+$$

β1w,xfn+1+β2w,xfn+2+β3w,xf[n+3],

(15)

where w is the frequence, $\beta_0(w,x), \beta_1(w,x)$, $\beta_2(w,x)$, and $\beta_3(w,x)$ are uninterrupted invariants. Consider [1], [11], [23]-[27], [29]-[30] for more info.

Forming the convergence criteria for Block Milne's device

To launch the Mathematica kernel programming codes of block Milne's device, the Adams-Bashforth $r-step$ method and Adams-Moulton $r-1-step$ are utilized as predictor-corrector pair possessing the like order. Merging [4], [10]-[11], [18]-[19], [29]-[30], block Milne's device shows that it is viable to estimate the principal local truncation error of the predictor-corrector pair in absence of estimating higher derivatives of $y(x)$. Presuming that $\tilde{p}=\bar{p}$, where $\bar{p}$ and $\tilde{p}$ establishes the order of the predictor and corrector pair. Instantly, for a method of order $\tilde{p}$, the investigation of the block Adams-Bashforth $r-step$ will bring forth the principal local truncation errors as

$$\tilde{C}_{p+6}^{[1]}h^{p+6}y^{(p+6)}(\tilde{x}_n)=y(x_{n+1})-y_{n+1}^{[l_1]}+O(h^{p+7}),$$
$$\tilde{C}_{p+6}^{[2]}h^{p+6}y^{(p+6)}(\tilde{x}_n)=y(x_{n+2})-y_{n+2}^{[l_2]}+O(h^{p+7}), \quad (16)$$
$$\tilde{C}_{p+6}^{[3]}h^{p+6}y^{(p+6)}(\tilde{x}_n)=y(x_{n+3})-y_{n+3}^{[l_3]}+O(h^{p+7}).$$

A like analysis of the block Adams-Moulton $r-1-step$ produces the principal local truncation errors as

$$\bar{C}_{p+6}^{[1]}h^{p+6}y^{(p+6)}(\bar{x}_n)=y(x_{n+1})-y_{n+1}^{[q_1]}+O(h^{p+7}),$$
$$\bar{C}_{p+6}^{[2]}h^{p+6}y^{(p+6)}(\bar{x}_n)=y(x_{n+2})-y_{n+2}^{[q_2]}+O(h^{p+7}), \quad (17)$$
$$\bar{C}_{p+6}^{[3]}h^{p+6}y^{(p+6)}(\bar{x}_n)=y(x_{n+3})-y_{n+2}^{[q_3]}+O(h^{p+7}),$$

where $\tilde{C}_{p+6}^{[1]}, \tilde{C}_{p+6}^{[2]}, \tilde{C}_{p+6}^{[1]}, \bar{C}_{p+6}^{[1]}, \bar{C}_{p+6}^{[2]}$ and $\bar{C}_{p+6}^{[3]}$ are in existence as an independent entity of the step-size h and $y(x)$ act as the result to the differential coefficient satisfying the initial consideration $y(x_n)\approx y_n$. Consider [4], [10]-[11], [18]-[19], [29]-[30] for more items.

To go forward, the presumption for small assesses of h is attained as

$$y^{(6)}(\tilde{x}_n)\approx y^{(6)}(\bar{x}_n),$$

and the implementation of the Mathematica kernel programming codes of block Milne's device banks directly on this condition.

Further reduction of the principal local truncation errors of (16) and (17) supra, in like manner, dropping terms of degree $O(h^{p+7})$, it becomes easier to achieve the computation of the principal local truncation errors of block Milne's device as

$$\bar{C}_{p+6}^{[1]}h^{p+6}y^{(p+6)}(\bar{x}_n)\approx\frac{\bar{C}_{p+6}^{[1]}}{\bar{C}_{p+6}^{[1]}-\tilde{C}_{p+6}^{[1]}}\left[y_{n+1}^{[l_1]}-y_{n+1}^{[q_1]}\right]<\varepsilon_1,$$

$$\bar{C}_{p+5}^{[2]}h^{p+6}y^{(p+6)}(\bar{x}_n)\approx\frac{\bar{C}_{p+6}^{[2]}}{\bar{C}_{p+6}^{[2]}-\tilde{C}_{p+6}^{[2]}}\left[y_{n+2}^{[l_2]}-y_{n+2}^{[q_2]}\right]<\varepsilon_2,$$

(18) $$\bar{C}_{p+6}^{[3]}h^{p+6}y^{(p+6)}(\bar{x}_n)\approx\frac{\bar{C}_{p+6}^{[3]}}{\bar{C}_{p+6}^{[3]}-\tilde{C}_{p+6}^{[3]}}\left[y_{n+3}^{[l_3]}-y_{n+3}^{[q_3]}\right]<\varepsilon_3.$$

Observing the statements that $y_{n+1}^{[l_1]}\neq y_{n+1}^{[q_1]}$, $y_{n+2}^{[l_2]}\neq y_{n+2}^{[q_2]}$ and $y_{n+3}^{[l_3]}\neq y_{n+3}^{[q_3]}$ are referred to as predicted and corrected approximates which are generated by the block Milne's device of order p, while $\bar{C}_{p+6}^{[1]}h^{p+6}y^{(p+6)}(\bar{x}_n)$, $\bar{C}_{p+6}^{[2]}h^{p+6}y^{(p+6)}(\bar{x}_n)$ and $\bar{C}_{p+6}^{[3]}h^{p+6}y^{(p+6)}(\bar{x}_n)$ are distinctly named the principal local truncation errors. $\varepsilon_1,\varepsilon_2$ and ε_3 are the limits of the convergence criteria of block Milne's device.

Nevertheless, the estimates of the principal local truncation error (18) is applied to settle whether to admit the results of the current step or to reconstruct the step with a slightly varying-step-size. The step is truly established on a trial run as defined by (18). Check [4], [10]-[11], [18]-[19], [29]-[30] for more details. The principal local truncation errors (18) is the convergence criteria of block Milne's, distinctly called the block Milne's device (estimate) for correcting to convergence.

IV. THE MATHEMATICA KERNEL PROGRAMMING CODES OF THE COMPUTED RESULTS AND DISCUSSION

In this section, the Mathematica kernel programming codes of the computational results demonstrate the execution of the block Milne's device employing the expounded trigonometrically fitted method for computing (1). The completed result supplied were obtained with the aid of Mathematica 9 kernel on Microsoft windows (64-bit) to showcase the accuracy and efficiency of the block Milne's device.

Some selected three tested problems were considered and worked out using **MKPC-BMD** at distinct convergence criteria of 10^{-4}, 10^{-6}, 10^{-8}, 10^{-10}, 10^{-11}, 10^{-12}, 10^{-13}, 10^{-14} and 10^{-15}. Check [3], [16], [23], [25]-[28] for more particulars. The Mathematica kernel programming codes of block Milne's device is composed applying Mathematica 9 kernel together with a written algorithm. This Mathematica kernel programming codes is implemented in a block by block manner as formulated by the block Milne's device. See appendix.

Tested problem 1: Consider the inhomogeneous IVP:
$y''(x)=-100y+99\sin(x), y(0)=1, \quad y'(0)=11,$
$0\le x\le 1000.$
Analytical Solution $y(x)=\cos(10x)+\sin(10x)+\sin(x)$
Tested problem 2: Consider the initial value ODE
$y''+\omega y=0, \quad y(0)=1, y'(0)2, \quad \omega=2.$
Analytical Solution: $y(x)=cos2x+sin2x.$
Tested problem 3: Consider the following mildly stiff IVP
$y''=-1001y'-1000y,\ y(0)=1, \quad y'(0)-1,$
$0\le x\le 10.$
Analytical Solution: $y(x)=e^{-x}.$

Table I, Table II and Table III- Shows the computational results of problems 1, 2, and 3 using MKPC-BMD compared with existing methods. The signifiers mentioned on Table I, Table II and Table III are expressed.

The terminology used is listed below:
MKPC-BMD: errors in MKPC-BMD (Mathematica kernel programming codes of block Milne's device) for tested problems 1, 2 and 3.
$M_{employed}$: method employed.
Max_{errors}: the magnitude of the maximum errors of MKPC-BMD
$C_{criteria}$: convergence criteria.
BHTRKNM: errors in BHTRKNM (block hybrid trigonometrically fitted Runge-Kutta-Nystrom method of $\delta=10^{-6}$) for tested problem 1. See (Nwange & Jator, 2017).

ISBN: 978-988-14048-5-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

BHT: errors in BHT (block hybrid trigonometrically fitted of $\delta = 10^{-6}$) for tested problem 1 and 3. See [27].
A(α)-S: errors in A(α)-S (an A(α)-stable method for solving initial value problems of ordinary differential equations) for tested problem 3. See [3].
TSDM: errors in TSDM (trigonometrically-fitted second derivative method) for tested problem 1. See [26].
FSBP-BCM: errors in FSBP-BCM (five steps block predictor-block corrector method for the solution of $y'' = f(x, y, y')$ for tested problem 2. See [28].
BHMTB: errors in BHMTB (block hybrid method with trigonometric basis) for tested problem 1. See [23].
HLMMs: errors in HLMMs (hybrid linear multistep methods) for tested problem 3. See [16].

Table I, Table II, and Table III displays the computational results for computing problems in the previous section employing MKPC-BMD.

Table I of problem 1

$\mathbf{M}_{\text{employed}}$	$\mathbf{Max}_{\text{errors}}$	$C_{criteria}$
TSDM	$2.5e-04$	10^{-4}
BHMTB	$6.8e-04$	
MKPC-BMD	$1.90254e-05$	10^{-4}
MKPC-BMD	$4.70666e-05$	
MKPC-BMD	$1.23255e-04$	
TSDM	$1.6e-06$	10^{-6}
BHMTB	$2.4e-06$	
BHT	$8.9e-06$	
BHTRKKNM	$1.26e-06$	
MKPC-BMD	$2.18794e-08$	10^{-6}
MKPC-BMD	$5.46559e-08$	
MKPC-BMD	$1.34927e-07$	
BHT	$4.2e-08$	10^{-8}
BHTRKKNM	$7.79e-08$	
MKPC-BMD	$3.7558e-11$	10^{-8}
MKPC-BMD	$9.38938e-11$	
MKPC-BMD	$8.98146e-11$	
MKPC-BMD	$1.94511e-13$	10^{-10}
MKPC-BMD	$4.84723e-13$	
MKPC-BMD	$3.77698e-13$	
MKPC-BMD	$2.22045e-15$	10^{-12}
MKPC-BMD	$4.44089e-15$	
MKPC-BMD	$3.77698e-13$	
MKPC-BMD	$2.22045e-15$	10^{-12}
MKPC-BMD	$4.44089e-15$	
MKPC-BMD	$4.21885e-15$	

Table II of problem 2

$\mathbf{M}_{\text{employed}}$	$\mathbf{Max}_{\text{errors}}$	$C_{criteria}$
A(α)-S	$4.3238e-11$	10^{-11}
MKPC-BMD	$9.6001e-12$	10^{-11}
MKPC-BMD	$2.40006e-11$	
MKPC-BMD	$2.88018e-11$	
FSBP-BCM	$3.397282e-13$	10^{-13}
FSBP-BCM	$4.54135e-13$	
FSBP-BCM	$5.193623e-13$	
FSBP-BCM	$5.194734e-13$	
MKPC-BMD	$9.63674e-14$	10^{-13}
MKPC-BMD	$2.40696e-13$	
MKPC-BMD	$2.8888e-13$	
MKPC-BMD	$1.33227e-15$	10^{-15}
MKPC-BMD	$3.10862e-15$	
MKPC-BMD	$2.22045e-15$	

Table III of problem 3

$\mathbf{M}_{\text{employed}}$	$\mathbf{Max}_{\text{errors}}$	$C_{criteria}$
BHT	$2.23e-04$	10^{-4}
HLMMs	$1.11852e-04$	
MKPC-BMD	$4.38159e-05$	10^{-4}
MKPC-BMD	$1.0825e-04$	
MKPC-BMD	$2.86048e-04$	
BHT	$3.36e-06$	10^{-6}
HLMMs	$1.68791e-06$	
MKPC-BMD	$4.7564e-08$	10^{-6}
MKPC-BMD	$1.18766e-07$	
MKPC-BMD	$3.30642e-07$	
BHT	$2.44e-08$	10^{-8}
HLMMs	$1.22041e-08$	
MKPC-BMD	$4.7956e-11$	10^{-8}
MKPC-BMD	$1.19876e-10$	
MKPC-BMD	$3.3546e-10$	
BHT	$1.96e-10$	10^{-10}
MKPC-BMD	$4.76286e-14$	10^{-10}
MKPC-BMD	$1.19238e-13$	
MKPC-BMD	$3.3662e-13$	
BHT	$2.13e-12$	10^{-12}
HLMMs	$1.06321e-12$	
MKPC-BMD	$3.33067e-16$	10^{-12}
MKPC-BMD	$5.55112e-16$	
MKPC-BMD	$2.22046e-16$	
BHT	$2.48e-14$	10^{-14}
HLMMs	$1.53766e-14$	
MKPC-BMD	$4.44089e-16$	10^{-14}
MKPC-BMD	$6.66134e-16$	
MKPC-BMD	$1.44329e-15$	

A written algorithm that will design a new step size and evaluate the maximum errors of the Mathematica kernel programming codes of block Milne's device is been prescribed as follows:

ISBN: 978-988-14048-5-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Step 1: Choose the step size for h
Step 2: The order of the block predictor-corrector pair must be the same
Step 3: The step number of the block predictor method must be one step greater than the block corrector method
Step 4: Define the convergence criteria of the block predictor-corrector pair
Step 5: Input the block predictor-corrector pair in any mathematical language
Step 6: Use any one step method to generate starting values if needed, if not, ignore step 6 and proceed to step 7
Step 7: Implement the block predictor-corrector pair in your chosen mathematical language
Step 8: If step 7 fails to converge, use this formula stated below to decide the appropriate step size for h to arrive at convergence and if not proceed to step 9

$$qh = \left| \frac{\varepsilon_1}{2\left(\bar{C}_{p+6}^{[1]} -, \bar{C}_{p+6}^{[1]}\right)} \right|^{\frac{1}{4}}$$

Step 9: Evaluate the maximum errors after convergence is attained
Step 10: Print maximum errors

V. CONCLUSION

Computational results have showed the MKPC-BMD is accomplished with the help of the convergence criteria. This convergence criteria determines whether the computational result is consented or reiterated. The computational results also establish the functioning of the MKPC-BMD is found to provide a better maximum errors than the TSDM, BHMTB, BHT, BHTRKNM, FSBP-BCM and A(α)-S at all tested convergence criteria of 10^{-4}, 10^{-6}, 10^{-8}, $10^{-10}, 10^{-11}$, 10^{-12}, 10^{-13}, 10^{-14} and 10^{-15} except for the convergence criteria of 10^{-4} (third iteration, tested problem 3) as cited in [3], [16], [23], [25]-[28]. Thusly, it can be resolved that the method formulated is worthy for computing special problems dealing with non-stiff, mildly stiff and stiff ODEs. Moreover, the MKPC-BMD is more effective compared to existing methods as stated supra for the reasons pointed out previously. Advance work will be to accomplish the MKPC-BMD on expounded exponentially fitted method.

ACKNOWLEDGEMENTS

The authors would like to appreciate Covenant University for providing financial backing through grants throughout the study period of time. Thanks to the anonymous reviewers for their continuous contribution.

REFERENCES

[1] M. L. Abell and J. P. Braselton "Mathematica by Example," 4th Edition, Elsevier, United State of America, 2009.
[2] A. O. Akinfenwa, S. N. Jator and N. M. Yao, "Continuous block backward differentiation formula for solving stiff ordinary differential equations," Computers and Mathematics with Applications, Vol. 65, No. 7, pp. 996- 1005, April 2013. 2013.
[3] T. A. Anake, S. A. Bishop, A. O. Adesanya, and M. C. Agarana, (2014) "An A(α)-stable method for solving initial value problems of ordinary differential equations. Advances in Differential Equations and Control Processes. Vol. 13, No. 1, pp. 21-35, May, 2014
[4] U. M. Ascher and L. R. Petzoid, Computer methods for ordinary differential equations and differential-algebraic equations. SIAM, United States of America, 1998
[5] J. G. Burkill, "Lectures on Approximation by polynomials," Tata Institute of Fundamental Research, Bombay, http://www.math.tifr.res.in/~publ/ln/tifr16.pdf,1959.
[6] M. Bond, "Convolutions and the Weierstrass approximation theorem," Department of Mathematics, Michigan State University, https://bondmatt.files.wordpress.com/2009/09/weierstrass2-01.pdf, 2009.
[7] M. Calvo, J. I. Montijano and M. Van Daele, "Exponentially fitted fifth-order two-step peer explicit methods," AIP Conference Proceedings, Vol. 1648, No. 1, pp. 1-4, 2015.
[8] R. D'Ambrosio, E. Esposito and B. Paternoster, "Exponentially fitted two- step hybrid methods for y^"=f(x,y)," Journal of Computational and Applied Mathematics, Vol. 235, pp. 4888-4897, 2011.
[9] R. D'Ambrosio, E. Esposito and B. Paternoster, "Exponentially fitted two- step Runge-Kutta methods," Applied Mathematics and Computation, Vol. 218, 7468-7480, 2012.
[10] J. R. Dormand, "Numerical methods for differential equations," A Computational Approach, London, 1996
[11] J. D. Faires and R. L. Burden, "Initial-value problems for ODEs," 3rd Edition, Dublin City University, Brooks Cole, 2012.
[12] W. Gautschi, "Numerical integration of ordinary differential equations based on trigonometric polynomials," Numerical Mathematics, Vol. 3, pp. 381-397, Received September, 1961.
[13] M. S. Gockenbach, "Mathematica tutorial,"2nd Edition, SIAM, www.math.mtu.edu/~msgocken/pdebook2/mathtut2.pdf, 2010.
[14] 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, March, 2007.
[15] Introduction to Mathematica-uc-davis economics, www.econ.ucdavis.edu/faculty/kdsalyer/LECTURES/Ecn200e/mathematica_intro.pdf, 2003.
[16] S. N. Jator, "On a class of hybrid methods for y^"=(x,y,y^')," International Journal of Pure and Applied Mathematics, Vol. 59, No. 4, 381-395, January, 2010.
[17] Y. L. Ken, I. F. Ismail and M. Suleiman, "Block methods for special second order odes," USA, 2011.
[18] J. D. Lambert, "Computational Methods in Ordinary Differential Equations," New York, 1973.
[19] J. D. Lambert, "Numerical Methods for Ordinary Differential Systems," New York, 1991.
[20] Z. A. Majid and M. B. Suleiman, "Implementation of Four-point fully implicit block method for solving ordinary differential equations. Applied Mathematics and Computation, Vol. 184, No. 2, pp. 514-522, January, 2007.
[21] Z. A. Majid M. Suleiman, "Parallel direct integration variable step block method for solving large system of higher Order odes," World Academy of Science, Engineering and Technology," Vol. 40, pp. 71-75, May, 2008.
[22] S. Mehrkanoon, Z. A. Majid and M. Suleiman, "A variable step implicit block multistep method for solving first-order ODEs." Journal of Comp. and Appl. Mathematics, Vol. 233, No. 9, pp. 2387-2394, March, 2010.
[23] F. F. Ngwane and S. N. Jator, "Block hybrid method using trigonometric basis for initial value problems with oscillating solutions," Numerical Algorithm, Vol. 63, pp. 713-725, October, 2012.
[24] F. F. Ngwane S. N. Jator, "Solving oscillatory problems using a block hybrid trigonometrically fitted method with two off-step points," Electronic Journal of Differential Equations, Conference on Differential Equations and Computational Simulations, Vol. 20, pp. 119-132, October, 2013.
[25] F. F. Ngwane and S. N. Jator, "Trigonometrically-fitted second derivative method for oscillatory problems," SpringerPlus, Vol. 3, No. 1, pp. 1-11, June, 2014.
[26] F. F. Ngwane and S. N. Jator, "Solving the telegraph and oscillatory differential equations by a block hybrid trigonometrically fitted

algorithm," Hindawi Publishing Corporation, Vol. 2015, pp. 1-15, October, 2015.

[27] F. F. Ngwane and S. N. Jator, "A trigonometrically fitted block method for solving oscillatory second-order initial value problems and Hamiltonian systems," Hindawi Publishing Corporation, Vol. 2017, pp. 1-14, January, 2017.

[28] M. R. Odekunle, M. O. Egwurube, A. O. Adesanya and M. O. Udo "Five steps block predictor-block corrector method for the solution of y^" f(x,〖 y,y〗 ^')," Applied Mathematics, Vol. 5, 1252-1266, May, 2014.

[29] J. G. Oghonyon, Okunuga, S. A. & S. A Iyase, (2016) "Milne's implementation on block predictor-corrector methods," Journal of Applied Sciences, Vol. 16, No. 5, pp. 236-241, April, 2016.

[30] J. G. Oghonyon, S. A. Okunuga and S. A. Bishop, "A Variable-step-size block predictor-corrector method for ODEs," Asian Journal of Applied Sciences, 10, No. 2, pp. 96-101, March, 2017.

[31] P. D. Polly, "Lecture 1- An introduction Mathematica A. key. Department of Geological Sciences," Indiana University, Available from http://www.indiana.edu/~g562/Lectures/Lecture%201%20Introduction %20to%20Mathematica%20A.pdf, 2012.

[32] A. Y. D. Siem, E. de Klerk and D. den Hertog, "Discrete Least-norm approximation by nonnegative (trigonometric) polynomials and rational functions," Tilburg University, Operations research. Vol. 2005, pp. 1-20, June, 2005.

APPENDIX

The Mathematica programming codes for solving problem 1, 2 and 3 using Block Milne's Device is written below.

```
g[t_]=Cos[10t]+Sin[10t]+Sin[t]
w=frequency
h==x[n]= given starting values
t=given values
g[1]=g[0]+h(g'[0])+(h^2/2)g"[0]+(h^3/6)g"'[0]+(h^4/24)g""[0]+(h^5/120)g""'[0]
g[2]=g[1]+h(g'[x[n]])+(h^2/2)g"[x[n]]+(h^3/6)g"'[x[n]]+(h^4/24)g""[x[n]]+(h^5/120)g""'[x[n]]
g[3]=g[2]+h(g'[x[n]+h])+(h^2/2)g"[x[n]+h]+(h^3/6)g"'[x[n]+h]+(h^4/24)g""[x[n]+h]+(h^5/120)g""'[x[n]+h]
g[4]=g[3]+h(g'[x[n]+2h])+(h^2/2)g"[x[n]+2h]+(h^3/6)g"'[x[n]+2h]+(h^4/24)g""[x[n]+2h]+(h^5/120)g""'[x[n]+2h]
g[5]=g[4]+h(g'[x[n]+3h])+(h^2/2)g"[x[n]+3h]+(h^3/6)g"'[x[n]+3h]+(h^4/24)g""[x[n]+3h]+(h^5/120)g""'[x[n]+3h]

t=x[n]+2h
g[4]=2g[3]-g[2]+h^2((43/40+1/w^5-1/w^4+7/(6w^2))g"[t]+(-11/60-3/w^5+3/w^4-3/w^2)g"[t-x[n]]+(17/120+3/w^5
-3/w^4+5/(2w^2))g"[t-x[n]+h]+(-1/30-1/w^5+1/w^4-2/(3w^2))g"[t-x[n]+2h])
t=x[n]+4h
g[6]=3g[4]-2g[3]+h^2((1697/360+16/w^5+61/w^4+16/(3w^2))g"[t]+(-101/30-48/w^5-149/w^4-14/w^2)g"[t-   x[n]]+(107/60+48/w^5+118/w^4+12/w^2)g"[t-
x[n]+h]+(-2/5-10/(3w^2)-87/(3w^4)-16/w^5)g"[t-x[n]+2h])
t=x[n]+6h
g[8]=4g[5]-3g[4]+h^2((4687/360+81/w^5+483/w^4+27/(2w^2))g"[t]+(-251/20-243/w^5-1201/w^4-36/w^2)g"[t-
x[n]]+(257/40+243/w^5+959/w^4+63/(2w^2))g"[t-x[n]+h]+(-43/30-81/w^5-239/w^4-9/w^2)g"[t-x[n]+2h])
t=x[n]+5h
g[7]=2g[6]-g[5]+h^2((43/40+1/w^5-1/w^4+7/(6w^2))g"[t]+(-11/60-3/w^5+3/w^4-3/w^2)g"[t-x[n]]+(17/120+3/w^5-3/w^4+5/(2w^2))g"[t-x[n]+h]+(-1/30-
1/w^5+1/w^4-2/(3w^2))g"[t-x[n]+2h])
t=x[n]+7h
g[9]=3g[7]-2g[6]+h^2((1697/360+16/w^5+61/w^4+16/(3w^2))g"[t]+(-101/30-48/w^5-149/w^4-14/w^2)g"[t-x[n]]+(107/60+48/w^5+118/w^4+12/w^2)g"[t-
x[n]+h]+(-2/5-10/(3w^2)-87/(3w^4)-16/w^5)g"[t-x[n]+2h])
t=x[n]+9h
g[11]=4g[8]-3g[7]+h^2((4687/360+81/w^5+483/w^4+27/(2w^2))g"[t]+(-251/20-243/w^5-1201/w^4-36/w^2)g"[t-
x[n]]+(257/40+243/w^5+959/w^4+63/(2w^2))g"[t-x[n]+h]+(-43/30-81/w^5-239/w^4-9/w^2)g"[t-x[n]+2h])
t=x[n]+8h

y[t_]=Cos[10t]+Sin[10t]+Sin[t]
w=frequency
h=x[n]= given starting values
t=given values
y[1]=y[0]+h(y'[0])+(h^2/2)y"[0]+(h^3/6)y"'[0]+(h^4/24)y""[0]+(h^5/120)y""'[0]
y[2]=y[1]+h(y'[x[n]])+(h^2/2)y"[x[n]]+(h^3/6)y"'[x[n]]+(h^4/24)y""[x[n]]+(h^5/120)y""'[x[n]]
y[3]=y[2]+h(y'[x[n]+h])+(h^2/2)y"[x[n]+h]+(h^3/6)y"'[x[n]+h]+(h^4/24)y""[x[n]+h]+(h^5/120)y""'[x[n]+h]
y[4]=y[3]+h(y'[x[n]+2h])+(h^2/2)y"[x[n]+2h]+(h^3/6)y"'[x[n]+2h]+(h^4/24)y""[x[n]+2h]+(h^5/120)y""'[x[n]+2h]
y[5]=y[4]+h(y'[x[n]+3h])+(h^2/2)y"[x[n]+3h]+(h^3/6)y"'[x[n]+3h]+(h^4/24)y""[x[n]+3h]+(h^5/120)y""'[x[n]+3h]

t=x[n]+2h
y[4]=2y[3]-y[2]+h^2((131/1200+1/(12w^2)+1/(10w^4)-1/(10w^5))y"[t-x[n]+h]+(39/20+1/w^5-1/w^4-1/(3w^2))y"[t+x[n]]+(-347/240+1/(4w^2)+3/(2w^4)-
3/(2w^5))y"[t+x[n]+h]+(29/75-3/(5w^4)+3/(5w^5))y"[t+x[n]+2h])
t=x[n]+4h
y[6]=3y[4]-2y[3]+h^2((81/600+4/(15w^2)+32/(5w^4)-8/(5w^5))y"[t-x[n]+h]+(157/30+16/w^5-33/w^4-2/(3w^2))y"[t+x[n]]+(-377/120-
24/w^5+34/w^4)y"[t+x[n]+h]+(58/75+2/(5w^2)-37/(5w^4)+48/w^5)y"[t+x[n]+2h])
t=x[n]+6h
y[8]=4y[5]-3y[4]+h^2((-127/1200+9/(20w^2)+487/(10w^4)-81/(10w^5))y"[t-x[n]+h]+(551/60+81/w^5-245/w^4)y"[t+x[n]]+(-307/80-9/(4w^2)+493/(2w^4)-
243/(2w^5))y"[t+x[n]+h]+(19/25+9/(5w^2)-251/(5w^4)+243/(5w^5))y"[t+x[n]+2h])
t=x[n]+5h
y[7]=2y[6]-y[5]+h^2((131/1200+1/(12w^2)+1/(10w^4)-1/(10w^5))y"[t-x[n]+h]+(39/20+1/w^5-1/w^4-1/(3w^2))y"[t+x[n]]+(-347/240+1/(4w^2)+3/(2w^4)-
3/(2w^5))y"[t+x[n]+h]+(29/75-3/(5w^4)+3/(5w^5))y"[t+x[n]+2h])
t=x[n]+7h
y[9]=3y[7]-2y[6]+h^2((81/600+4/(15w^2)+32/(5w^4)-8/(5w^5))y"[t-x[n]+h]+(157/30+16/w^5-33/w^4-2/(3w^2))y"[t+x[n]]+(-377/120-
24/w^5+34/w^4)y"[t+x[n]+h]+(58/75+2/(5w^2)-37/(5w^4)+48/w^5)y"[t+x[n]+2h])
t=x[n]+9h
y[11]=4y[8]-3y[7]+h^2((-127/1200+9/(20w^2)+487/(10w^4)-81/(10w^5))y"[t-x[n]+h]+(551/60+81/w^5-245/w^4)y"[t+x[n]]+(-307/80-
9/(4w^2)+493/(2w^4)-243/(2w^5))y"[t+x[n]+h]+(19/25+9/(5w^2)-251/(5w^4)+243/(5w^5))y"[t+x[n]+2h])
```